

**UNIVERSIDAD AUTONOMA DE MADRID**

**ESCUELA POLITECNICA SUPERIOR**



## **TRABAJO FIN DE GRADO**

**Gestión de Trabajos en Movilidad (GTM)  
(COSITI\_120)**

**Alba Calvo Ruiz**

**Junio 2013**



# **Gestión de Trabajos en Movilidad (GTM)**

**AUTOR: Alba Calvo Ruiz**  
**TUTOR: Álvaro Ortigosa Juárez**

**Dpto. de Ingeniería Informática**  
**Escuela Politécnica Superior**  
**Universidad Autónoma de Madrid**  
**Junio de 2013**



## **RESUMEN**

El proyecto GTM (Gestión de Trabajos en Movilidad) es una aplicación de gestión para empresas con actividad de SAT (Servicio de Atención Técnica) con soporte de movilidad.

GTM gestiona el ciclo completo de vida de un aviso incluyendo aspectos como la recepción, asignación y envío a técnicos, cumplimentación del trabajo realizado, etc.

La aplicación proporciona servicios de movilidad. Los técnicos reciben automáticamente los avisos en sus terminales móviles, desde donde graban a su vez el trabajo realizado a la finalización del aviso.

Además, GTM pone a disposición de la empresa la visualización de mapas relativos a la ubicación y rutas de sus técnicos.

La implementación constituye un sistema integrado por una aplicación de gestión de avisos, una Web de recepción y cumplimentación de trabajos por parte de los técnicos y una solución de geolocalización y representación cartográfica compuesta por una aplicación de tracking GPS para terminales móviles y una Web de presentación de mapas.

**Palabras clave:** movilidad, geolocalización, tracking GPS, terminal móvil, gestión de avisos.



## **ABSTRACT**

The GTM (Gestión de Trabajos en Movilidad) project is a management application for companies providing mobile Technical Assistance Service (TAS).

GTM manages the complete cycle of a notice including aspects such as reception, assignment and dispatch of technicians, fulfilment of a task, etc.

The application provides mobility services. Technicians receive tasks automatically on their mobile terminals, where they can also save the task accomplished once finished.

In addition, GTM provides the company with visual maps with the position and routes of its technicians.

The implementation forms an integrated system made up of a task management application, a Web of reception and fulfilment of the tasks carried out by technicians. It also provides a georeference and mapping representation formed by a GPS tracking application for mobile terminals and Web map representation.

**Keywords:** mobility, georeference, GPS tracking, mobile terminal, notice management service.





## ***Agradecimientos***

En primer lugar, quiero dar las gracias a mis padres Miguel y Paz por todo lo que han hecho por mí y por su apoyo incondicional en todo momento. Os agradezco mucho todo el tiempo y esfuerzo que habéis dedicado a cuidarme anteponiendo siempre mis necesidades a las vuestras. Gracias por haberme ayudado a llegar hasta aquí.

Una mención especial merece mi hermanito ya no tan pequeño Miguel, que es la alegría de la casa y la persona más amable que conozco. Gracias por haberme hecho compañía durante todos estos años.

También quiero dar las gracias a mis abuelos por preocuparse tanto por mí y a mi tía Chary por encender velas a la virgen en épocas de exámenes y por haber sido como mi segunda madre.

Me gustaría también dar las gracias a Víctor Álvarez Gutiérrez por su ayuda y por haber compartido sus conocimientos conmigo.

Por último, pero no menos importante, también quiero dar las gracias a todos los profesores que me han formado y en especial a mi tutor Álvaro Ortigosa por el tiempo que ha dedicado a revisar y corregir mi trabajo.



## INDICE DE CONTENIDOS

Glosario .....	7
1 Introducción.....	9
1.1 Motivación.....	9
1.2 Objetivos.....	11
1.3 Organización de la memoria.....	12
2 Estado del arte .....	13
2.1 Introducción.....	13
2.2 Aplicación cliente de escritorio .....	13
2.2.1 Java .....	13
2.2.1.1 Ventajas .....	13
2.2.1.2 Desventajas .....	13
2.2.2 .NET.....	13
2.2.2.1 Ventajas .....	13
2.2.2.2 Desventajas .....	14
2.3 Aplicación Web .....	14
2.3.1 Java (Java Server Pages).....	14
2.3.1.1 Ventajas .....	14
2.3.1.2 Desventajas .....	14
2.3.2 ASP.NET .....	14
2.3.2.1 Ventajas .....	14
2.3.2.2 Desventajas .....	15
2.3.3 PHP .....	15
2.3.3.1 Ventajas .....	15
2.3.3.2 Desventajas .....	15
2.4 Plataformas de referencia para terminales en movilidad.....	15
2.4.1 Windows phone .....	15
2.4.1.1 Ventajas .....	15
2.4.1.2 Desventajas .....	15
2.4.2 Blackberry OS.....	16
2.4.2.1 Ventajas .....	16
2.4.2.2 Desventajas .....	16
2.4.3 iOS .....	16
2.4.3.1 Ventajas .....	16
2.4.3.2 Desventajas .....	16
2.4.4 Android .....	17
2.4.4.1 Ventajas .....	17
2.4.4.2 Desventajas .....	17
2.5 Base de Datos. ....	17
2.5.1 SQLServer .....	17
2.5.1.1 Ventajas .....	17
2.5.1.2 Desventajas .....	18

2.5.2 MySQL .....	18
2.5.2.1 Ventajas .....	18
2.5.2.2 Desventajas .....	18
2.5.3 PostgreSQL .....	18
2.5.3.1 Ventajas .....	18
2.5.3.2 Desventajas .....	18
3 Diseño .....	19
3.1 Componentes del proyecto. Análisis de tecnologías .....	19
3.1.1 Base de datos SQLServer .....	19
3.1.2 Aplicación cliente Windows Forms .....	20
3.1.3 Aplicación Web en PHP .....	22
3.1.4 Aplicación cliente de tracking en Android .....	22
3.2 Arquitectura .....	22
3.2.1 Base de datos .....	23
3.2.2 Aplicación cliente de gestión .....	24
3.2.3 Aplicación Web .....	32
3.2.4 Aplicación de geolocalización .....	36
4 Desarrollo .....	39
4.1 Base de datos .....	39
4.1.1 Tecnologías empleadas .....	39
4.2 Aplicación cliente de gestión .....	39
4.2.1 Tecnologías empleadas .....	39
4.2.2 Estructura .....	39
4.2.2.1 Negocio y ModeloDTO .....	40
4.2.2.1.1 MovilidadPresentacion .....	43
4.2.3 Diagramas .....	43
4.2.3.1 Máquina de estados .....	43
4.2.3.1.1 Administrador .....	43
4.2.3.1.2 Empleado .....	44
4.2.3.2 Diagrama UML .....	44
4.2.3.3 Diagrama de secuencia .....	46
4.3 Aplicación Web .....	47
4.3.1 Tecnologías empleadas .....	47
4.3.2 Estructura .....	47
4.3.3 Diagramas .....	50
4.3.3.1 Máquina de estados .....	50
4.4 Aplicación de geolocalización .....	51
4.4.1 Tecnologías empleadas .....	51
4.4.2 Estructura .....	51
4.4.3 Diagramas .....	53
4.4.3.1 Máquina de estados .....	53
4.4.3.2 Diagrama UML .....	53
4.4.3.3 Diagrama de secuencia .....	54
5 Integración, pruebas y resultados .....	55
5.1 Pruebas modulares .....	55

5.1.1 Pruebas de aplicación cliente de gestión.....	55
5.1.1.1 Entorno de prueba.....	55
5.1.1.2 Pruebas realizadas.....	55
5.1.1.2.1 Pruebas de correcto almacenamiento en la base de datos .	55
5.1.1.2.1.1. Diseño de la prueba .....	55
5.1.1.2.1.2. Análisis de los resultados .....	56
5.1.1.2.2 Pruebas de interfaz .....	61
5.1.1.2.2.1. Diseño de la prueba .....	61
5.1.1.2.2.2. Análisis de resultados.....	61
5.1.2 Pruebas de aplicación Web .....	62
5.1.2.1 Entorno de prueba.....	62
5.1.2.2 Pruebas realizadas.....	62
5.1.2.2.1 Pruebas de flujo.....	62
5.1.2.2.1.1. Diseño de la prueba .....	62
5.1.2.2.1.2. Análisis de resultados.....	62
5.1.2.2.2 Pruebas de notificación de errores .....	62
5.1.2.2.2.1. Diseño de la prueba .....	62
5.1.2.2.2.2. Análisis de resultados.....	62
5.1.3 Pruebas de aplicación de geolocalización.....	63
5.1.3.1 Entorno de prueba.....	63
5.1.3.2 Pruebas realizadas.....	64
5.1.3.2.1 Pruebas de acceso a servicios Web .....	64
5.1.3.2.1.1. Diseño de la prueba .....	64
5.1.3.2.1.2. Análisis de resultados.....	64
5.1.3.2.2 Pruebas de servicio de geolocalización.....	64
5.1.3.2.2.1. Diseño de la prueba .....	64
5.1.3.2.2.2. Análisis de resultados.....	65
5.2 Pruebas de integración.....	66
5.2.1 Diseño de la prueba.....	66
5.2.2 Análisis de resultados .....	66
6 Conclusiones y trabajo futuro.....	67
6.1 Conclusiones.....	67
6.2 Trabajo futuro .....	68
Bibliografía.....	71
Anexos.....	I
A Manual de instalación .....	I
B Manual de usuario.....	II

# INDICE DE FIGURAS

FIGURA 1. ARQUITECTURA DE UNA APLICACIÓN WEB. ....	20
FIGURA 2. ARQUITECTURA DE UNA APLICACIÓN CLIENTE.....	21
FIGURA 3. ESQUEMA DEL PROYECTO GTM. ....	22
FIGURA 4. ESQUEMA DE LA BASE DE DATOS. ....	23
FIGURA 5. APLICACIÓN CLIENTE DE GESTIÓN: PANTALLA INICIAL. ....	24
FIGURA 6. APLICACIÓN CLIENTE DE GESTIÓN: PANTALLA DE LOGIN. ....	24
FIGURA 7. APLICACIÓN CLIENTE DE GESTIÓN: MENÚ DE EMPLEADO. ....	25
FIGURA 8. APLICACIÓN CLIENTE DE GESTIÓN: MENÚ DE ADMINISTRADOR. ....	25
FIGURA 9. APLICACIÓN CLIENTE DE GESTIÓN: VENTANA DE CLIENTES. ....	26
FIGURA 10. APLICACIÓN CLIENTE DE GESTIÓN: ALTA DE AVISO.....	26
FIGURA 11. APLICACIÓN CLIENTE DE GESTIÓN: REJILLA DE AVISOS CON CAMBIOS REFLEJADOS... ..	27
FIGURA 12. APLICACIÓN CLIENTE DE GESTIÓN: NOTIFICACIÓN EN EL CORREO DE LA EMPRESA. ...	27
FIGURA 13. APLICACIÓN CLIENTE DE GESTIÓN: MENSAJE DE NOTIFICACIÓN DE RECEPCIÓN DE AVISO. ....	27
FIGURA 14. APLICACIÓN CLIENTE DE GESTIÓN: VENTANA DE EMPLEADOS.....	28
FIGURA 15. APLICACIÓN CLIENTE DE GESTIÓN: ALTA DE EMPLEADO. ....	28
FIGURA 16. APLICACIÓN DE GESTIÓN: REJILLA DE EMPLEADOS CON CAMBIOS REFLEJADOS. ....	29
FIGURA 17. APLICACIÓN CLIENTE DE GESTIÓN: REJILLA DE USUARIO CON CAMBIOS REFLEJADOS.....	29
FIGURA 18. APLICACIÓN CLIENTE DE GESTIÓN: VENTANA DE TRACKING DE TÉCNICOS. ....	29
FIGURA 19. APLICACIÓN CLIENTE DE GESTIÓN: UBICACIÓN DE TÉCNICOS. ....	30
FIGURA 20. APLICACIÓN CLIENTE DE GESTIÓN: RUTA DE TÉCNICOS 1.....	30
FIGURA 21. APLICACIÓN CLIENTE DE GESTIÓN: RUTA DE TÉCNICOS 2.....	31
FIGURA 22. APLICACIÓN CLIENTE DE GESTIÓN: MODIFICACIÓN DE USUARIO.....	31
FIGURA 23. APLICACIÓN CLIENTE DE GESTIÓN: MODIFICACIÓN DE DATOS DE USUARIO.....	31
FIGURA 24. APLICACIÓN WEB: LOGIN.....	32
FIGURA 25. APLICACIÓN WEB: MENÚ DE ADMINISTRADOR.....	32
FIGURA 26. APLICACIÓN WEB: RUTA DE TÉCNICO.....	33
FIGURA 27. APLICACIÓN WEB: UBICACIÓN DE TÉCNICOS.....	33
FIGURA 28. APLICACIÓN WEB: MENÚ DE TÉCNICO.....	33
FIGURA 29. APLICACIÓN WEB: AVISOS NUEVOS.....	34
FIGURA 30. APLICACIÓN WEB: DATOS DE AVISO.....	34
FIGURA 31. APLICACIÓN WEB: AVISOS PENDIENTES.....	34
FIGURA 32. APLICACIÓN WEB: DATOS DE AVISO CON OPCIÓN DE CUMPLIMENTACIÓN.....	35
FIGURA 33. APLICACIÓN WEB: CUMPLIMENTACIÓN DE AVISO.....	35
FIGURA 34. APLICACIÓN WEB: AVISOS REALIZADOS EL ÚLTIMO MES.....	35
FIGURA 35. APLICACIÓN ANDROID: LOGIN.....	36
FIGURA 36. APLICACIÓN ANDROID: HABILITACIÓN DE GPS.....	37
FIGURA 37. APLICACIÓN ANDROID: MENÚ.....	37
FIGURA 38. APLICACIÓN ANDROID: SERVICIO DE TRACKING.....	38
FIGURA 39: CAPAS DE LA APLICACIÓN CLIENTE.....	39
FIGURA 40. APLICACIÓN CLIENTE DE GESTIÓN: MÁQUINA DE ESTADOS DE ADMINISTRADOR.....	43
FIGURA 41. APLICACIÓN CLIENTE DE GESTIÓN: MÁQUINA DE ESTADOS DE EMPLEADO.....	44
FIGURA 42. APLICACIÓN CLIENTE DE GESTIÓN: DIAGRAMA UML.....	44
FIGURA 43. APLICACIÓN CLIENTE DE GESTIÓN: FORMULARIO MDI 1.....	45
FIGURA 44. APLICACIÓN CLIENTE DE GESTIÓN: FORMULARIO MDI 2.....	45
FIGURA 45. APLICACIÓN CLIENTE DE GESTIÓN: DIAGRAMA DE SECUENCIA.....	46

FIGURA 46. APLICACIÓN WEB: MÁQUINA DE ESTADOS.....	50
FIGURA 47. APLICACIÓN ANDROID: MÁQUINA DE ESTADOS.....	53
FIGURA 48. APLICACIÓN ANDROID: DIAGRAMA UML.....	53
FIGURA 49. APLICACIÓN ANDROID: DIAGRAMA DE SECUENCIA. ....	54
FIGURA 50. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN CLIENTE: REJILLA DE EMPLEADOS. .	56
FIGURA 51. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN CLIENTE: CONSULTA DE EMPLEADOS. .....	56
FIGURA 52. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN CLIENTE: CONSULTA DE USUARIOS.	56
FIGURA 53. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN CLIENTE: ALTA DE EMPLEADO. ....	57
FIGURA 54. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN CLIENTE: CONSULTA DE EMPLEADOS CON CAMBIOS REFLEJADOS.....	57
FIGURA 55. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN CLIENTE: CONSULTA DE USUARIOS CON CAMBIOS REFLEJADOS.....	57
FIGURA 56. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN CLIENTE: REJILLA DE EMPLEADOS CON CAMBIOS REFLEJADOS. ....	58
FIGURA 57. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN CLIENTE: REJILLA DE USUARIOS CON CAMBIOS REFLEJADOS. ....	58
FIGURA 58. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN CLIENTE: MODIFICACIÓN DE EMPLEADO. ....	58
FIGURA 59. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN CLIENTE: CONSULTA DE EMPLEADOS CON CAMBIOS REFLEJADOS 2.....	58
FIGURA 60. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN CLIENTE: CONSULTA DE USUARIOS CON CAMBIOS REFLEJADOS 2.....	59
FIGURA 61. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN CLIENTE: REJILLA DE EMPLEADOS CON CAMBIOS REFLEJADOS 2. ....	59
FIGURA 62. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN CLIENTE: REJILLA DE USUARIOS CON CAMBIOS REFLEJADOS 2. ....	59
FIGURA 63. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN CLIENTE: ELIMINACIÓN DE EMPLEADO. ....	59
FIGURA 64. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN CLIENTE: CONSULTA DE EMPLEADOS CON CAMBIOS REFLEJADOS 3.....	60
FIGURA 65. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN CLIENTE: CONSULTA DE USUARIOS CON CAMBIOS REFLEJADOS 3.....	60
FIGURA 66. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN CLIENTE: REJILLA DE EMPLEADOS CON CAMBIOS REFLEJADOS 3. ....	60
FIGURA 67. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN CLIENTE: REJILLA DE USUARIOS CON CAMBIOS REFLEJADOS 3. ....	60
FIGURA 68. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN CLIENTE: PRUEBA DE CAMPOS SIN RELLENAR. ....	61
FIGURA 69. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN CLIENTE: PRUEBA DE CAMPOS DEMASIADO LARGOS.....	61
FIGURA 70. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN WEB: PRUEBA DE CAMPOS SIN RELLENAR EN EL FORMULARIO DE LOGIN.....	62
FIGURA 71. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN WEB: PRUEBA LOGIN INCORRECTO.	63
FIGURA 72. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN WEB: PRUEBA DE AUSENCIA DE AVISOS SIN LEER EN LA BASE DE DATOS. ....	63
FIGURA 73. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN DE GEOLOCALIZACIÓN: PRUEBA DE LOGIN.....	64
FIGURA 74. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN DE GEOLOCALIZACIÓN: PRUEBA DE SERVICIO DE GEOLOCALIZACIÓN. ....	65

FIGURA 75. PRUEBAS MODULARES. PRUEBA DE APLICACIÓN DE GEOLOCALIZACIÓN: CONSULTA DE COORDENADAS DE GEOLOCALIZACIÓN ALMACENADAS EN LA BASE DE DATOS. ....	65
FIGURA 76. ANEXO I: MANUAL DE INSTALACIÓN. PAQUETE DE INSTALACIÓN DE LA APLICACIÓN CLIENTE DE GESTIÓN. ....	IV
FIGURA 77. ANEXO I: MANUAL DE INSTALACIÓN. ACCESOS DIRECTOS DE LA APLICACIÓN CLIENTE DE GESTIÓN. ....	IV
FIGURA 78. ANEXO I: MANUAL DE INSTALACIÓN. INSTALACIÓN DE LA APLICACIÓN ANDROID. ....	V
FIGURA 79. ANEXO II: MANUAL DE USUARIO. ESQUEMA DEL PROYECTO GTM. ....	III
FIGURA 80. ANEXO II: MANUAL DE USUARIO. APLICACIÓN DE GESTIÓN: VENTANA DE LOGIN. ....	IV
FIGURA 81. ANEXO II: MANUAL DE USUARIO. APLICACIÓN DE GESTIÓN: MENÚ DE ADMINISTRADOR. ....	IV
FIGURA 82. ANEXO II: MANUAL DE USUARIO. APLICACIÓN DE GESTIÓN: VENTANA DE CLIENTES. .	V
FIGURA 83. ANEXO II: MANUAL DE USUARIO. APLICACIÓN DE GESTIÓN: VENTANA DE EMPLEADOS. ....	VI
FIGURA 84. ANEXO II: MANUAL DE USUARIO. APLICACIÓN DE GESTIÓN: VENTANA DE TÉCNICOS. ....	VII
FIGURA 85. ANEXO II: MANUAL DE USUARIO. APLICACIÓN DE GESTIÓN: VENTANA DE TRACKING. ....	VII
FIGURA 86. ANEXO II: MANUAL DE USUARIO. APLICACIÓN DE GESTIÓN: UBICACIÓN DE TÉCNICOS. ....	VIII
FIGURA 87. ANEXO II: MANUAL DE USUARIO. RUTA DE TÉCNICOS 1. ....	IX
FIGURA 88. ANEXO II: MANUAL DE USUARIO. APLICACIÓN DE GESTIÓN: RUTA DE TÉCNICOS 2. ....	IX
FIGURA 89. ANEXO II: MANUAL DE USUARIO. APLICACIÓN DE GESTIÓN: VENTANA DE USUARIOS. X	
FIGURA 90. ANEXO II: MANUAL DE USUARIO. APLICACIÓN DE GESTIÓN: VENTANA DE DATOS PERSONALES DE ADMINISTRADOR. ....	X
FIGURA 91. ANEXO II: MANUAL DE USUARIO. APLICACIÓN DE GESTIÓN: MENÚ DE EMPLEADO. ....	XI
FIGURA 92. ANEXO II: MANUAL DE USUARIO. APLICACIÓN DE GESTIÓN: VENTANA DE DATOS PERSONALES DE EMPLEADO. ....	XII
FIGURA 93. ANEXO II: MANUAL DE USUARIO. APLICACIÓN WEB: LOGIN. ....	XII
FIGURA 94. ANEXO II: MANUAL DE USUARIO. APLICACIÓN WEB: MENÚ DE ADMINISTRADOR. ...	XIII
FIGURA 95. ANEXO II: MANUAL DE USUARIO. APLICACIÓN WEB: RUTA DE TÉCNICOS. ....	XIII
FIGURA 96. ANEXO II: MANUAL DE USUARIO. APLICACIÓN WEB: UBICACIÓN DE TÉCNICOS. ....	XIV
FIGURA 97. ANEXO II: MANUAL DE USUARIO. APLICACIÓN WEB: MENÚ DE TÉCNICO. ....	XIV
FIGURA 98. ANEXO II: MANUAL DE USUARIO. APLICACIÓN WEB: AVISOS NUEVOS. ....	XV
FIGURA 99. ANEXO II: MANUAL DE USUARIO. APLICACIÓN WEB: AVISOS PENSIENTES. ....	XV
FIGURA 100. ANEXO II: MANUAL DE USUARIO. APLICACIÓN WEB: DATOS DE AVISO CON OPCIÓN DE CUMPLIMENTACIÓN. ....	XVI
FIGURA 101. ANEXO II: MANUAL DE USUARIO. APLICACIÓN WEB: CUMPLIMENTACIÓN DE AVISO. ....	XVI
FIGURA 102. ANEXO II: MANUAL DE USUARIO. APLICACIÓN WEB: AVISOS REALIZADOS EL ÚLTIMO MES. ....	XVI
FIGURA 103. ANEXO II: MANUAL DE USUARIO. APLICACIÓN ANDROID: LOGIN. ....	XVII
FIGURA 104. ANEXO II: MANUAL DE USUARIO. APLICACIÓN ANDROID: MENÚ / HABILITACIÓN DE GPS. ....	XVII
FIGURA 105. ANEXO II: MANUAL DE USUARIO. APLICACIÓN ANDROID: HABILITACIÓN DE GPS. ....	XVIII



# Glosario

---

**API** *Application Programming Interface*. Interfaz de programación de aplicaciones, es el conjunto de funciones y procedimientos que ofrece una biblioteca para ser utilizado por otro software como una capa de abstracción.

**Backup (o copia de seguridad)**. Copia de los datos originales que se realiza con el fin de disponer de un medio de recuperarlos en caso de pérdida.

**DAO** *Data Access Object*. Componente de software que suministra una interfaz común entre la aplicación y uno o más dispositivos de almacenamiento de datos, tales como una base de datos o un archivo.

**DTO** *Data Transfer Object*. Patrón de diseño utilizado para transferir datos entre subsistemas de aplicaciones de software. Los DTOs se usan habitualmente junto con los DAOs para recuperar datos de la base de datos.

**ETL** *Extract, Transform and Load*. Proceso que permite mover datos desde múltiples fuentes, reformatearlos y limpiarlos, y cargarlos en otra base de datos.

**Framework**. Estructura conceptual y tecnológica de soporte definido que puede servir de base para la organización y desarrollo de software. Típicamente, puede incluir soporte de programas, bibliotecas, y un lenguaje interpretado, entre otras herramientas, para así ayudar a desarrollar y unir los diferentes componentes de un proyecto.

**GPS**. *Global Position System*. Es un sistema global de navegación por satélite que permite determinar la posición de un objeto en todo el mundo.

**Hosting (o alojamiento Web)**. Servicio que provee a los usuarios de Internet un sistema para poder almacenar información, imágenes, vídeo, o cualquier contenido accesible vía Web.

**JSON**. *JavaScript Object Notation*. Formato ligero de intercambio de datos, completamente independiente del lenguaje de programación.

**LINQ**. *Language Integrated Query*. Componente de la plataforma Microsoft .NET que agrega capacidades de consulta a datos de manera nativa a los lenguajes .NET.

**MDI** *Multiple Document Interface*. Interfaz cuyas ventanas se encuentran dentro de una ventana padre (normalmente con la excepción de las ventanas modales).

**ORM** *Object-Relational Mapping*. Técnica de programación para convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y el utilizado en una base de datos relacional, utilizando un motor de persistencia.

**Polling**. Operación de consulta constante para crear una actividad sincrónica sin el uso de interrupciones.

**Push email.** Sistema de email en el que el nuevo email es transferido a su llegada al servidor de correo.

**SAT.** *Servicio de Atención Técnica.* Departamento o empresa encargada del mantenimiento o reparación de productos.

**Smartphone (o teléfono inteligente).** Teléfono móvil construido sobre una plataforma informática móvil, con una mayor capacidad de almacenar datos y realizar actividades semejantes a una mini computadora y conectividad que un teléfono móvil convencional.

**SSH.** *Secure Shell.* Nombre de un protocolo y del programa que lo implementa. Sirve para acceder a máquinas remotas a través de una red.

**TIC.** *Tecnologías de la Información y la Comunicación.* Conjunto de recursos, procedimientos y técnicas usadas en el procesamiento, almacenamiento y transmisión de información.

# 1 Introducción

---

## 1.1 Motivación

El concepto de movilidad hace referencia a un conjunto de tecnologías que permiten el acceso a la información en cualquier momento y desde cualquier lugar.

Actualmente, gracias a la gran popularización de las tecnologías móviles con la llegada de los “smartphones”, la movilidad se está convirtiendo en un tema crucial que afecta cada vez a más sectores: servicio y soporte al cliente, compras y ventas, logística e inventario, herramientas corporativas... [1] [2]

El impacto de la movilidad en empresas es cada vez más evidente. Cada vez son más las empresas preocupadas por la toma de decisiones en tiempo real y el control de la productividad del personal fuera de la oficina que optan por integrar la movilidad en su ámbito profesional. [3]

Según IDC (International Data Corporation), en 2013 más de 1.190 millones de empleados de todo el mundo utilizarán la tecnología móvil, lo que representa un 34,9 % de la población activa.<sup>1</sup>

Pero no todas las empresas disponen de los mecanismos necesarios para la gestión de movilidad en sus entornos TIC. En particular, numerosas empresas de pequeño tamaño no tienen implantado ningún sistema de gestión de trabajos en movilidad. El proyecto GTM (Gestión de Trabajos en Movilidad) se propone cubrir esta necesidad. Aunque el entorno de pequeñas empresas presenta la dificultad de la escasa capacidad de gasto en infraestructuras, cuenta por otra parte con la ventaja del elevado número de potenciales clientes.

El proyecto GTM está enfocado a empresas que realizan trabajos de mantenimiento y reparación en domicilios de clientes. Tradicionalmente, la gestión de los avisos se realizaba mediante la emisión de partes en papel y no se tenía control de la actividad de los técnicos salvo por el informe que éstos emitían al final del día. GTM propone una solución que cubre toda la vida del aviso contemplando diferentes hitos como la recepción, asignación y comunicación del aviso al técnico, grabación del trabajo realizado desde terminales en movilidad, consulta de trabajos pendientes-realizados, etc. Además, facilita una herramienta de geolocalización y supervisión de rutas de técnicos mediante tracking GPS de terminales y presentación de resultados utilizando Google Maps.

GTM podría ser el embrión de futuros desarrollos de una aplicación sectorial dirigida a empresas SAT (electricidad, telecomunicación, ascensores, etc).

---

<sup>1</sup> "Worldwide Mobile Worker Population 2009-2013 Forecast" (IDC, 2009) ["Previsión de los trabajadores móviles en todo el mundo 2009-2013" (IDC, 2009)]

Mi motivación nace de mi interés en el área de la gestión empresarial y en la adquisición de conocimientos relacionados con las múltiples tecnologías implicadas.

Gracias a la realización de este trabajo he podido desarrollar habilidades en los siguientes ámbitos:

- Contratación y gestión de un hosting donde poder alojar los elementos del proyecto (servidor SQLServer y servidor Web Microsoft-IIS/7.0 con soporte de PHP).
- Creación y mantenimiento de la base de datos del proyecto utilizando la herramienta Management Studio 2012<sup>2</sup> contra el servidor de base de datos SQLServer<sup>3</sup>.
- Desarrollo de aplicación cliente Windows Forms<sup>4</sup> en C# con las siguientes características:
  - Utilización de controles avanzados de interfaz de usuario de Infragistics<sup>5</sup>.
  - Interoperabilidad con base de datos SQLServer mediante el ORM (Object-relational mapping) Microsoft Entity Framework<sup>6</sup>.
- Desarrollo de aplicación Web en lenguaje PHP<sup>7</sup> con las siguientes características:
  - Interoperabilidad con SQLServer.
  - Uso de Javascript<sup>8</sup> y librerías del mismo como jQuery<sup>9</sup>.
  - Servicio Web para recepción de tracking GPS y anotación en base de datos SQLServer.
  - Representación cartográfica de localización y rutas de técnicos mediante el empleo de la API de Google Maps<sup>10</sup>.
- Desarrollo de aplicación cliente para dispositivos Android con las siguientes características:
  - Tracking GPS.
  - Almacenamiento de resultados de tracking en servidor SQLServer mediante peticiones a servicio Web.
  - Comportamiento multihilo (tareas asíncronas y servicios en Android).
- Creación de paquetes de instalación.

---

<sup>2</sup> [http://en.wikipedia.org/wiki/SQL\\_Server\\_Management\\_Studio](http://en.wikipedia.org/wiki/SQL_Server_Management_Studio)

<sup>3</sup> <http://www.microsoft.com/es-es/sqlserver/default.aspx>

<sup>4</sup> <http://msdn.microsoft.com/es-es/library/dd30h2yb.aspx>

<sup>5</sup> <http://www.infragistics.com/>

<sup>6</sup> <http://msdn.microsoft.com/es-es/library/bb399572.aspx>

<sup>7</sup> <http://php.net/>

<sup>8</sup> <http://www.w3schools.com/js/>

<sup>9</sup> <http://jquery.com/>

<sup>10</sup> <https://developers.google.com/maps/?hl=es>

## 1.2 Objetivos

El objetivo del proyecto GTM es dar respuesta a las necesidades de gestión de empresas de mantenimiento y reparación en domicilios de clientes. Por ejemplo:

- Recepción de avisos de clientes con asignación del técnico en el momento de la recepción o con posterioridad.
- Introducción del trabajo a realizar desde una aplicación cliente en terminales situados en las dependencias de la empresa.
- Notificación inmediata de avisos a técnicos.
- Consulta de avisos pendientes y finalización de los mismos (grabación de trabajo realizado, etc) por parte de los técnicos desde terminales en movilidad (teléfonos con conectividad a Internet y navegador).
- Recepción y grabación automática en la base de datos de la realización de trabajos introducida por los técnicos.
- Representación cartográfica de ubicación y rutas de los técnicos.

La implementación de una respuesta a estas necesidades permitiría, entre otras cosas:

- Eliminar las tareas de emisión y reparto de partes de trabajo en papel.
- Conocer en tiempo real la realización de los trabajos, con la consiguiente ventaja de cara a la asignación de nuevos trabajos a técnicos, etc.
- Facilitar a los receptores de avisos información actualizada útil en su interacción con los clientes, permitiendo respuestas como "el técnico ya acudió hace una hora pero no había nadie en casa", etc.
- Evitar la grabación de datos por el personal de la oficina con el consiguiente ahorro de tiempo y eliminación de errores derivados de la interpretación de los textos manuscritos de los técnicos.
- Asignación directa de avisos entrantes a técnicos próximos con el consecuente ahorro de tiempo, especialmente en avisos de mantenimiento donde no se cobra el desplazamiento.
- Reducción del absentismo derivado del control de rutas de técnicos.

### **1.3 Organización de la memoria**

La memoria consta de seis capítulos y dos anexos que se detallan a continuación:

- **Capítulo 1: Introducción**  
Explicación de la motivación y objetivos del proyecto. También se detalla la estructura de la memoria.
- **Capítulo 2: Estado del arte**  
Descripción del estado actual de las tecnologías implicadas en el proyecto, incluyendo una comparativa de sus principales ventajas y desventajas.
- **Capítulo 3: Diseño**  
Resumen de la arquitectura del proyecto y breve explicación de cada una de sus partes, incluyendo detalles de su interfaz.
- **Capítulo 4: Desarrollo**  
Análisis de las tecnologías implicadas en la realización del proyecto y breve explicación del funcionamiento del mismo. Se incorporan además diagramas (máquina de estados, UML, secuencia) para ilustrar mejor la explicación.
- **Capítulo 5: Integración, pruebas y resultados**  
Detalle de las pruebas realizadas (modulares, de integración, etc.).
- **Capítulo 6: Conclusiones y trabajo futuro**  
Exposición resumida del planteamiento y logro de los objetivos del proyecto y de su posible ampliación.
- **Anexo I: Manual de instalación**  
Explicación del procedimiento de instalación.
- **Anexo II: Manual de usuario**  
Explicación de la utilización de la solución por parte del usuario.

## 2 Estado del arte

---

### 2.1 Introducción

En los apartados siguientes se analiza la tecnología disponible en relación con los componentes del proyecto detallados en el capítulo 3 "Diseño".

En concreto se contempla la situación de las tecnologías relacionadas con:

- Desarrollo de aplicaciones cliente de escritorio.
- Desarrollo de aplicaciones Web.
- Aplicaciones cliente para terminales en movilidad.
- Servidores de bases de datos.

### 2.2 Aplicación cliente de escritorio

Una aplicación de escritorio puede ser desarrollada con múltiples tecnologías: C, C++, Delphi, Java, .NET (C# u otros lenguajes), Python, etc.

Las dos tecnologías a la cabeza actualmente son Java y .NET. [4]

A continuación se describen ambas opciones:

#### 2.2.1 Java

##### 2.2.1.1 Ventajas

- Dispone de varias APIs de desarrollo gráfico: AWT (Abstract Window Toolkit) con librerías Swing, SWT (Standard Widget Toolkit).<sup>11</sup>
- Alta portabilidad de sus aplicaciones.
- Es una plataforma libre.

##### 2.2.1.2 Desventajas

- Tiene una mayor orientación a desarrollos WEB frente a .NET más orientado a aplicaciones cliente, presentando una menor productividad que .NET en el desarrollo de interfaces de usuario en el ámbito de aplicaciones cliente para PCs.
- El diseño de interfaces gráficas con AWT y Swing no resulta trivial.

#### 2.2.2 .NET

##### 2.2.2.1 Ventajas

- Dispone de varias tecnologías de creación de interfaces: Windows Forms, Windows Presentation Foundation.<sup>12</sup>

---

<sup>11</sup> [http://es.wikipedia.org/wiki/Java\\_\(lenguaje\\_de\\_programaci%C3%B3n\)#En\\_aplicaciones\\_de\\_escritorio](http://es.wikipedia.org/wiki/Java_(lenguaje_de_programaci%C3%B3n)#En_aplicaciones_de_escritorio)

<sup>12</sup> [http://msdn.microsoft.com/es-es/library/w0x726c2\(v=vs.100\).aspx](http://msdn.microsoft.com/es-es/library/w0x726c2(v=vs.100).aspx)

- Tiene una fuerte orientación al desarrollo de aplicaciones cliente para PC, presentando una alta productividad en este entorno.

#### **2.2.2.2 Desventajas**

- Actualmente presenta poca portabilidad de aplicaciones, aunque se están llevando a cabo proyectos como “Mono”<sup>13</sup> para hacer .NET compatible con otros sistemas operativos.
- Es una plataforma de desarrollo propietaria que debe ser licenciada bajo pago.

## **2.3 Aplicación Web**

Las tecnologías líderes en el desarrollo de aplicaciones Web son Java, ASP.NET y PHP.<sup>14</sup>  
[5]

A continuación se analizan las ventajas y desventajas de cada una de ellas:

### **2.3.1 Java (Java Server Pages<sup>15</sup>).**

#### **2.3.1.1 Ventajas**

- Es una plataforma muy potente, con multitud de frameworks como Spring, Struts, etc.<sup>16</sup>
- Es la solución con mayor implantación, sobre todo en grandes desarrollos Web.
- Existe máquina virtual para múltiples sistemas operativos.
- Es orientado a objetos.
- Presenta soporte de acceso a base de datos mediante la API JDBC (Java DatabaseConnectivity).<sup>17</sup>
- Es una plataforma libre.

#### **2.3.1.2 Desventajas**

- Como corresponde a la gran potencia de esta plataforma la curva de aprendizaje es elevada frente a productos más ligeros como PHP.
- Puede no existir JDBC para bases de datos poco comerciales.

### **2.3.2 ASP.NET**

#### **2.3.2.1 Ventajas**

- Dispone de un framework de gran funcionalidad, equiparable a Java en aspectos como: XML, acceso a datos, carga de archivos, generación de imágenes, control de rendimiento y operaciones de registro, etc.<sup>18</sup>

<sup>13</sup>[http://www.mono-project.com/Main\\_Page](http://www.mono-project.com/Main_Page)

<sup>14</sup>[http://es.wikipedia.org/wiki/Desarrollo\\_web](http://es.wikipedia.org/wiki/Desarrollo_web)

<sup>15</sup><http://www.oracle.com/technetwork/java/javasee/jsp/index.html>

<sup>16</sup>[http://www.ehow.com/about\\_6190145\\_java-spring-vs\\_-struts.html](http://www.ehow.com/about_6190145_java-spring-vs_-struts.html)

<sup>17</sup>[http://es.wikipedia.org/wiki/Java\\_Database\\_Connectivity](http://es.wikipedia.org/wiki/Java_Database_Connectivity)

<sup>18</sup>[http://en.wikipedia.org/wiki/.NET\\_Framework#Design\\_features](http://en.wikipedia.org/wiki/.NET_Framework#Design_features)



- Detecta el tipo de navegador del cliente a la hora de realizar una petición al servidor y determina la versión de HTML que éste soporte.

### **2.3.2.2 Desventajas**

- Dependencia del sistema operativo Windows
- Menor implantación en la comunidad de desarrolladores que Java.
- Es una plataforma de pago.

## **2.3.3 PHP**

### **2.3.3.1 Ventajas**

- Presenta una curva de aprendizaje rápida
- Es un lenguaje multiplataforma
- Ofrece soporte para una gran cantidad de bases de datos: Oracle, MySQL, SQLServer, PostgreSQL, etc.<sup>19</sup>

### **2.3.3.2 Desventajas**

- Menos orientado a grandes desarrollos que las plataformas anteriores.
- De forma nativa no ofrece las mismas posibilidades que Java y .NET en cuanto a modularización y organización en capas de la aplicación.

## **2.4 Plataformas de referencia para terminales en movilidad.**

Las cuatro plataformas de mayor implantación son: [6]

- Android de Google
- iOS de Apple
- Blackberry OS de Blackberry (antes RIM)
- Windows Phone de Microsoft

A continuación se analiza en detalle cada una de ellas:

### **2.4.1 Windows phone**

#### **2.4.1.1 Ventajas**

- Utiliza una potente herramienta de desarrollo basada en .NET Compact Framework.<sup>20</sup>

#### **2.4.1.2 Desventajas**

- Es una incógnita si alcanzará un grado de operatividad, eficiencia en el aprovechamiento del hardware e implantación suficiente.

<sup>19</sup> <http://php.net/manual/es/refs.database.php>

<sup>20</sup> [https://es.wikipedia.org/wiki/Windows\\_Phone#Desarrollo\\_de\\_aplicaciones](https://es.wikipedia.org/wiki/Windows_Phone#Desarrollo_de_aplicaciones)

## 2.4.2 Blackberry OS

### 2.4.2.1 Ventajas

- Proporciona servicios avanzados de comunicaciones basados en plataforma propietaria.

Especialmente interesante resulta el servicio de “Push email”<sup>21</sup>, que proporciona inmediatez en el correo entrante sin necesidad de realización de “polling” (consulta constante) desde el terminal. [7]

### 2.4.2.2 Desventajas

- “Push email”, a pesar de ser un gran servicio, va perdiendo peso frente al conjunto de servicios avanzados ofrecidos por otros tipos de terminal. Por otra parte, la recepción inmediata de emails mediante “polling” utilizando otros terminales ofrece un funcionamiento correcto sin incremento de coste (el consumo de ancho de banda no es significativo).
- El futuro de la empresa está sujeto a cambios e incertidumbres.

Recientemente se ha producido un cambio de plataforma (Blackberry 10) y de herramienta de desarrollo (Java RIM hacia C++).<sup>22</sup>

## 2.4.3 iOS

### 2.4.3.1 Ventajas

- Es líder junto a Android en prestaciones y disponibilidad de aplicaciones, así como en implantación y número de usuarios.<sup>23</sup>

### 2.4.3.2 Desventajas

- Es una solución cerrada con terminales de Apple. Esto puede ser un problema para servicios en movilidad que requieran tipos específicos de terminales (de bajo coste, con equipamiento específico como lectores de código de barras hardware, impresoras...).
- Es necesario certificar todas las aplicaciones en cada una de sus versiones e incluirlas en el AppStore de Apple. Esto puede ser un inconveniente para pequeños desarrollos.
- Utiliza un lenguaje de desarrollo específico y poco extendido (ObjectiveC) con la correspondiente curva de aprendizaje.<sup>24</sup>

---

<sup>21</sup> [http://es.wikipedia.org/wiki/Tecnolog%C3%ADA\\_Push](http://es.wikipedia.org/wiki/Tecnolog%C3%ADA_Push)

<sup>22</sup> <http://alt1040.com/2013/02/thorsten-heins-y-la-era-blackberry>

<sup>23</sup> <http://www.24horas.cl/tendencias/mundodigital/iphone-lidera-en-eeuu-y-android-es-el-lider-mundial-481300>

<sup>24</sup> [http://es.wikipedia.org/wiki/Apple\\_iOS#Contenido\\_del\\_SDK](http://es.wikipedia.org/wiki/Apple_iOS#Contenido_del_SDK)

## 2.4.4 Android

### 2.4.4.1 Ventajas

- Es líder junto a iPhone en prestaciones y disponibilidad de aplicaciones así como en implantación y número de usuarios (92% de los nuevos smartphones en España funcionan con Android<sup>25</sup>).
- Su sistema operativo es de libre distribución y se encuentra implantado en una gama muy amplia de terminales. Esto puede ser una ventaja significativa en servicios profesionales que requieran terminales de bajo coste, con hardware específico, etc.
- Utiliza un lenguaje de desarrollo Java con una curva de aprendizaje rápida y entornos de desarrollo de libre distribución.

### 2.4.4.2 Desventajas

- Debido a la proliferación de fabricantes que lo implementan aparece el problema de "fragmentación de terminales" debido a los cambios introducidos en el sistema operativo por algunos de estos fabricantes. Esto provoca que los programas desarrollados para un modelo de terminal no sean totalmente compatibles con el resto.<sup>26</sup>

## 2.5 Base de Datos.

Existen múltiples servidores de bases de datos en el mercado actual.

Por coste y tamaño quedan fuera del ámbito de este proyecto productos como DB2, Oracle, etc.

A continuación se evalúan varios servidores de gran implantación, que presentan un coste asumible (en algunos casos nulo por tratarse de productos de carácter gratuito) y una capacidad de almacenamiento acorde a las necesidades de GTM: SQLServer, MySql, y PostgreSQL. [8]

### 2.5.1 SQLServer

#### 2.5.1.1 Ventajas

- Se trata de un producto consolidado acompañado de una serie de herramientas de mantenimiento y productividad que aportan un gran valor añadido como:
  - Cubos OLAP, etc a la altura de grandes bases de datos.<sup>27</sup>
  - Management Studio con soporte de tareas de mantenimiento de la base de datos, de backup, etc.
  - Agente de programación de tareas (de backup, etc.).
  - Sistema de integración de datos externos (SSIS).

<sup>25</sup> [http://cincodias.com/cincodias/2013/04/17/tecnologia/1366205805\\_952527.html](http://cincodias.com/cincodias/2013/04/17/tecnologia/1366205805_952527.html)

<sup>26</sup> <http://www.gacetatecnologica.com/movilizate/?p=236>

<sup>27</sup> [http://es.wikipedia.org/wiki/Cubo\\_OLAP](http://es.wikipedia.org/wiki/Cubo_OLAP)

- Transact SQL.<sup>28</sup>
- Integración con herramientas Microsoft (.NET, etc).
- Facilidad de aprendizaje.

#### **2.5.1.2 Desventajas**

- Es un producto de pago que es necesario licenciar.

### **2.5.2 MySQL.**

#### **2.5.2.1 Ventajas**

- Es un producto consolidado de gran rapidez.
- Libre distribución.

#### **2.5.2.2 Desventajas**

- Su adquisición por parte de Oracle condiciona su futuro a las posibles decisiones de esta corporación.<sup>29</sup> Esto está provocando el desplazamiento de usuarios a otras alternativas como MariaDB. [9]
- No cuenta con herramientas de mantenimiento similares en productividad a Management Studio de SQLServer. Tampoco cuentan con herramientas ETL (Extract, Transform, Load)<sup>30</sup> para la extracción de datos de fuentes externas, reformato y carga en la base de datos, mientras que SQLServer cuenta con SSIS (SQLServerIntegrationServices).<sup>31</sup>

### **2.5.3 PostgreSQL**

#### **2.5.3.1 Ventajas**

- Se trata de un producto consolidado y es una alternativa real a MySQL y SQLServer.
- Libre distribución.

#### **2.5.3.2 Desventajas**

- Herramientas menos amigables que las de SQLServer y MySQL.
- .NET posee una fuerte integración con el ORM Entity Framework para SQLServer con mucha documentación y ejemplos disponibles. No he encontrado referencias a ORMs para PostgreSQL integrables con .NET con un grado de difusión y documentación equiparable al de Entity Framework.

<sup>28</sup> <http://es.wikipedia.org/wiki/Transact-SQL>

<sup>29</sup> <http://www.espaciolinux.com/2010/01/aprobada-compra-sun-oracle-mysql-java-parte-trato/>

<sup>30</sup> [http://es.wikipedia.org/wiki/Extract,\\_transform\\_and\\_load](http://es.wikipedia.org/wiki/Extract,_transform_and_load)

<sup>31</sup> <http://msdn.microsoft.com/en-us/library/ms141026.aspx>

## 3 Diseño

---

### 3.1 Componentes del proyecto. Análisis de tecnologías

El proyecto GTM (Gestión de Trabajos en Movilidad) incluye los siguientes componentes fundamentales:

- Aplicación cliente de gestión instalada en PCs de usuarios
- Aplicación Web para el intercambio de avisos en movilidad con los técnicos y para la presentación de mapas de ubicación y rutas de técnicos.
- Aplicación cliente de tracking GPS instalada en el terminal móvil de cada técnico.

Además, da soporte a la consulta y almacenamiento de información en una base de datos común, mediante un servidor de base de datos al que realizan peticiones los componentes citados.

Para la elección de la tecnología de cada uno de los componentes y del servidor de base de datos se han tenido en cuenta las opciones predominantes en el mercado actual identificadas en el capítulo 2 “Estado del arte”.

A continuación se detallan las tecnologías seleccionadas así como el criterio utilizado durante el proceso de evaluación:

#### 3.1.1 Base de datos SQLServer

Se ha descartado la opción de MySQL debido a las incertidumbres derivadas de su adquisición por Oracle. En cuanto a la alternativa MariaDB, aunque en rápido crecimiento, se ha desestimado por no ser aún un producto consolidado.

Aunque se trata de un producto de pago que es necesario licenciar, se ha optado por SQLServer frente a PostgreSQL teniendo en cuenta las ventajas que ofrece en cuanto a consolidación y herramientas de mantenimiento y productividad más amigables, así como por su rápida curva de aprendizaje.

Su elección ha permitido la utilización del ORM (Object-relationalmapping) Microsoft Entity Framework de alta productividad para permitir la interoperabilidad entre la aplicación cliente .NET y la base de datos.

Para empresas de pequeño tamaño sería posible utilizar en un principio el producto SQLExpress de carácter gratuito que actualmente cuenta con un límite de almacenamiento de 10 GB.<sup>32</sup>

---

<sup>32</sup><http://www.microsoft.com/es-es/download/details.aspx?id=35579>

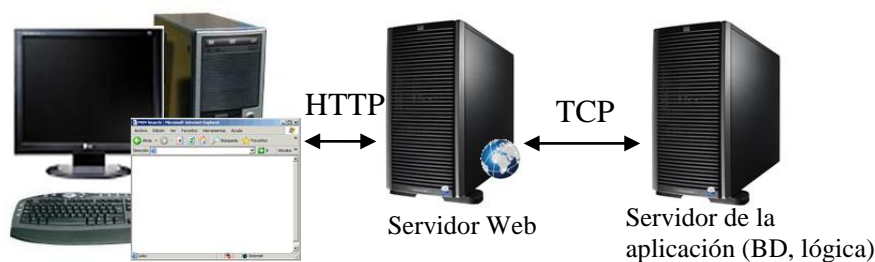
### 3.1.2 Aplicación cliente Windows Forms

La aplicación da soporte a la gestión de la empresa, manteniendo los datos almacenados en la base de datos a través de una interfaz de usuario.

Además de la opción de aplicación cliente se ha considerado la posibilidad de desarrollo de una aplicación Web. Durante el proceso de evaluación se han tenido en cuenta las siguientes características de cada solución: [10] [11]

- Aplicación Web:

En esta opción el usuario final interactúa por medio de un navegador Web con la aplicación localizada en el servidor, donde residen los datos y la lógica de la aplicación.



**Figura 1. Arquitectura de una aplicación Web.**

La aplicación Web presenta las ventajas inherentes a este tipo de aplicación como son la actualización instantánea sin necesidad de actuar en los PCs del usuario y el acceso desde cualquier ubicación utilizando un terminal dotado de navegador. Esta ventaja es determinante para implantaciones en grandes organizaciones con un elevado número de usuarios.

En el caso concreto del proyecto GTM, el posible soporte futuro de la actividad completa de una empresa SAT puede conducir a una interfaz de usuario compleja. En esta situación la productividad en el desarrollo sería superior en el entorno de aplicación cliente.

- Aplicación Cliente:

En esta opción la lógica se desplaza al cliente, que asume todas las funcionalidades de la aplicación, dejando al servidor la gestión de las transacciones contra la base de datos.



**Figura 2. Arquitectura de una aplicación cliente.**

Esta opción presenta la ventaja de una alta productividad a la hora de construir interfaces con un elevado número de formularios o con un comportamiento sofisticado de los mismos (comportamiento MDI donde las ventanas hijas residen bajo una única ventana padre que se encarga de mantener el estado de las mismas<sup>33</sup>, navegación entre formularios, ventanas maestro/detalle, etc.).

El principal inconveniente es la necesidad de actuar en el PC de cada usuario para instalar y actualizar la aplicación. Esto puede ser un gran problema en grandes organizaciones con un elevado número de usuarios.

En la comparación de tecnologías se ha tenido en cuenta la posible evolución futura de la aplicación con vistas al soporte de la gestión completa de la actividad SAT de una empresa con la proliferación y complejidad de formularios que ello podría suponer. Por otra parte, se ha tenido en cuenta que el proyecto va dirigido a empresas de tamaño pequeño o mediano más que a grandes estructuras lo cual limita el problema de la implantación de nuevas versiones en múltiples PCs de usuario.

Teniendo en cuenta las consideraciones anteriores, se ha optado por la opción de aplicación cliente.

Al realizar esta selección se ha pensado en un escenario de trabajo consistente en una aplicación cliente atacando a un servidor de base de datos ubicado en las propias dependencias de la empresa. La implantación del escenario de pruebas del proyecto no se ajusta a esta situación, ya que la base de datos SQLServer se encuentra ubicada en un hosting Web. Esta configuración provisional obedece a la necesidad de poder realizar pruebas y demostraciones con terminales atacando a un servidor de base de datos publicado en la Web.

Como lenguaje de desarrollo para la aplicación cliente se ha optado por C# sobre la plataforma .NET. Se ha tenido en cuenta que la plataforma de desarrollo de Microsoft tiene una fuerte implantación y orientación a aplicaciones cliente.

Dentro del entorno .NET, se ha optado por Windows Forms frente a Windows PresentationFoundation (WPF) a pesar de las avanzadas prestaciones de esta última opción como formularios escalables, separación entre diseño y programación, etc. En la decisión ha prevalecido la alta productividad de Windows Forms frente a las elevadas prestaciones de WPF, sobre todo teniendo en cuenta que se trata de una aplicación de uso interno.

---

<sup>33</sup>[http://en.wikipedia.org/wiki/Multiple\\_document\\_interface](http://en.wikipedia.org/wiki/Multiple_document_interface)

### 3.1.3 Aplicación Web en PHP

Debido a la envergadura del proyecto, se ha optado por la opción de PHP para desarrollar la aplicación Web.

PHP ofrece grandes ventajas como su rápida curva de aprendizaje y su soporte para la utilización de la base de datos SQLServer.

### 3.1.4 Aplicación cliente de tracking en Android

Todas las plataformas consideradas en el capítulo 2 “Estado del arte” incorporan funcionalidad GPS, por lo que este aspecto no ha sido decisivo a la hora de seleccionar uno de ellos.

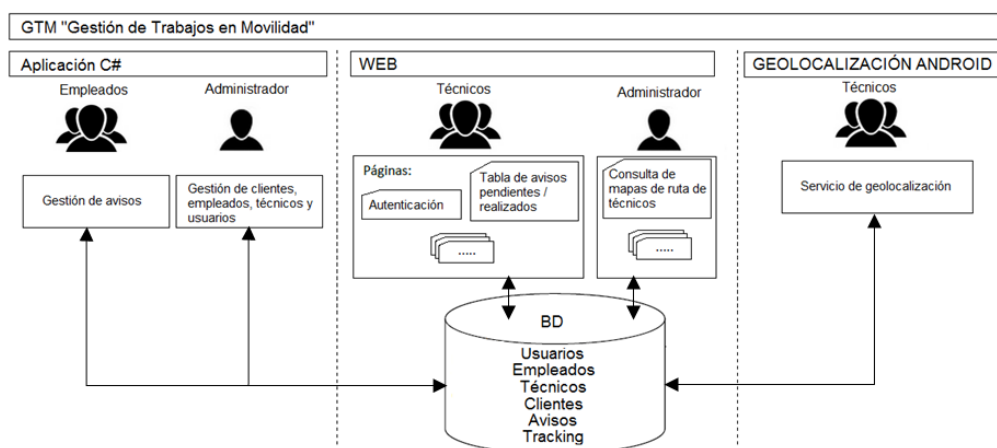
De acuerdo con las características expuestas en el capítulo (líder en prestaciones, alta implantación, sistema operativo de libre distribución, lenguaje de desarrollo Java), se ha seleccionado la plataforma Android como la más idónea para el desarrollo del proyecto. No obstante, dado que la aplicación cliente encargada de la subida de coordenadas geográficas realiza esta tarea mediante peticiones a un servicio Web, el proyecto está abierto a cambios del tipo de plataforma.

## 3.2 Arquitectura

El diseño del proyecto está compuesto de tres partes fundamentales:

- Aplicación cliente Windows Forms
- Aplicación Web PHP
- Aplicación de geolocalización GPS para plataformas Android

Todas ellas acceden a una base de datos común, como se muestra en el siguiente esquema:



**Figura 3. Esquema del proyecto GTM.**

A continuación se describe más detalladamente cada una de las partes, así como la estructura de la base de datos utilizada.

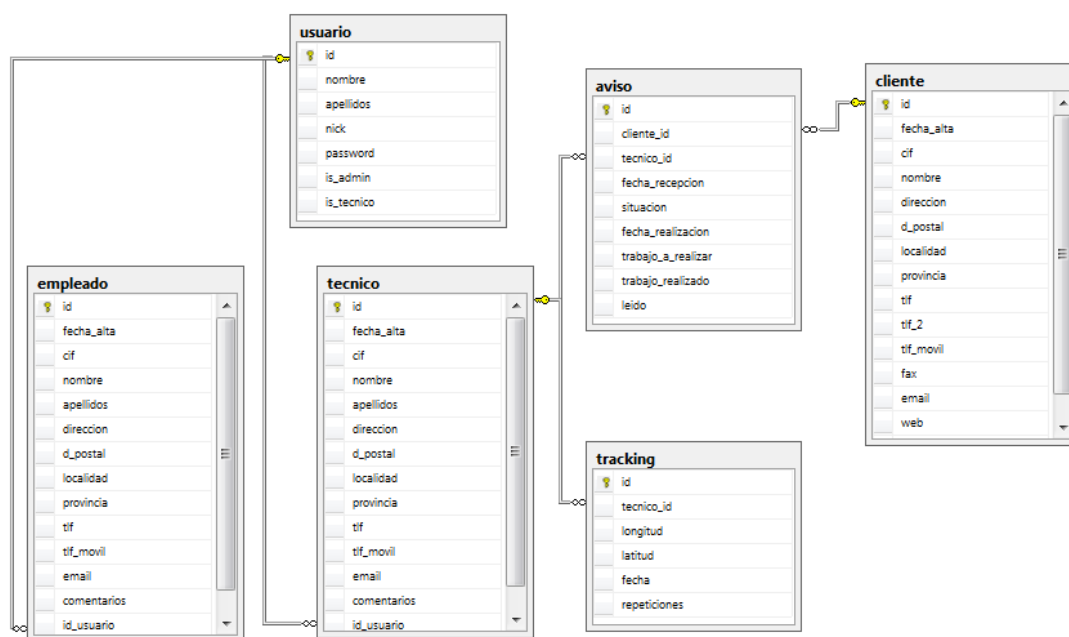


### 3.2.1 Base de datos

GTM cuenta con una base de datos alojada en un servidor SQL Server que permite el alta, la modificación y la eliminación de usuarios, empleados, técnicos, clientes y avisos desde la aplicación cliente de gestión.

También posibilita a los técnicos la visualización y la cumplimentación de avisos en la aplicación Web, así como el envío y almacenamiento de coordenadas de localización desde la aplicación ejecutada en sus terminales móviles para posibilitar la posterior consulta de su ubicación y rutas.

La base de datos cuenta con la siguiente estructura de tablas:



**Figura 4. Esquema de la base de datos.**

Se puede observar que hay tres roles de usuario: empleado, técnico y administrador (empleado con el flag is\_admin=true). Estos roles permiten la identificación selectiva de usuarios dependiendo de la aplicación a la que acceden. Así, la aplicación cliente sólo puede ser utilizada por usuarios con rol de empleado (ya sea administrador o no), la página Web sólo puede ser utilizada por el administrador y los técnicos, diferenciando contenido según el rol, y la aplicación de geolocalización sólo puede ser utilizada por técnicos.

El esquema también incluye la tabla aviso, asociada con un técnico y un cliente determinados.

También hay una tabla tracking que almacena las coordenadas de geolocalización de un técnico enviadas desde su terminal móvil.

### 3.2.2 Aplicación cliente de gestión

El objetivo de la aplicación es la gestión de usuarios de la base de datos, así como el alta de avisos y su asignación a técnicos.

La aplicación contempla dos roles de usuario:

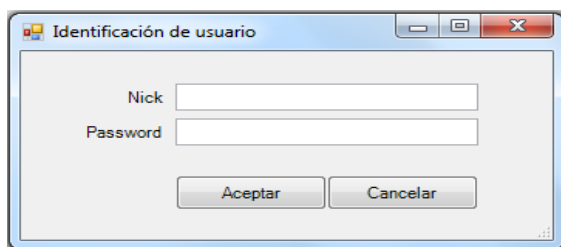
- El administrador, encargado de gestionar (dar de alta, modificar o eliminar) clientes, empleados, técnicos y usuarios. Además, puede visualizar la ubicación y rutas de los técnicos.
- El empleado, con un tipo de acceso más restringido y con la única posibilidad de gestionar avisos y visualizar ubicación y rutas de técnicos.

En esta aplicación se darán de alta los avisos, que se notificarán al técnico mediante el envío de un correo electrónico a su cuenta de la empresa. Estos avisos podrán ser visualizados y gestionados posteriormente por los técnicos mediante la página Web.

A continuación se expone el diagrama de flujo de la aplicación:



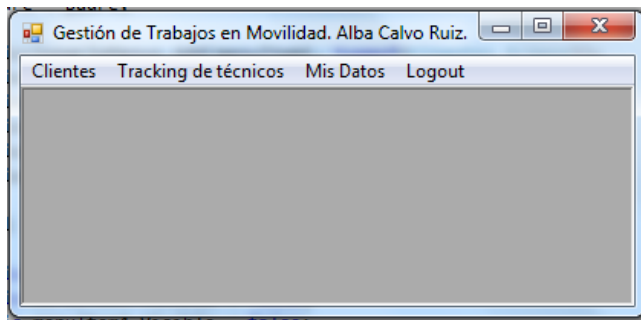
**Figura 5. Aplicación cliente de gestión: Pantalla inicial.**



**Figura 6. Aplicación cliente de gestión: Pantalla de login.**

Inicialmente se muestra la pantalla de login. Si los datos no son correctos se muestra un mensaje de error y si lo son el flujo del programa puede seguir dos caminos según el tipo de usuario autenticado:

- Empleado



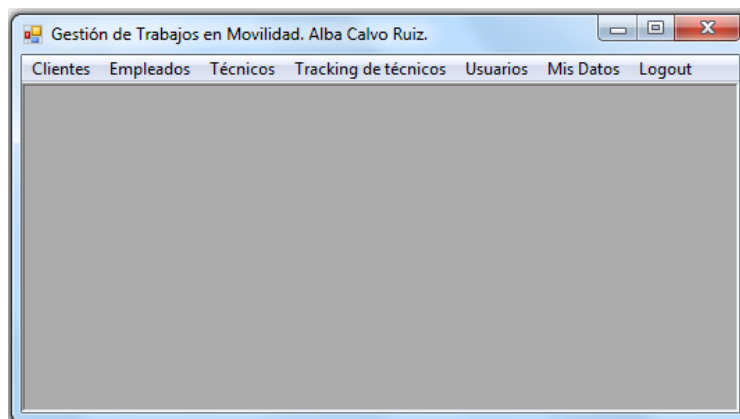
**Figura 7. Aplicación cliente de gestión: Menú de empleado.**

El empleado dispone de varias opciones:

- “Clientes”: gestión de avisos asociados a clientes.
- “Tracking de técnicos”: consulta de ubicación y rutas de técnicos.
- “Mis Datos”: cambio de contraseña.
- “Logout”: cierre de sesión.

Toda esta funcionalidad está comprendida dentro de la del usuario con rol de administrador, por lo que se describirá en detalle más adelante.

- Administrador



**Figura 8. Aplicación cliente de gestión: Menú de administrador.**

El administrador dispone de varias opciones:

- “Clientes”: gestión de avisos asociados a clientes.
- “Empleados”: gestión de empleados de la empresa.
- “Técnicos”: gestión de técnicos de la empresa.
- “Tracking de técnicos”: consulta de ubicación y rutas de técnicos.
- “Usuarios”: gestión de usuarios de la solución GTM.
- “Mis Datos”: cambio de contraseña.
- “Logout”: cierre de sesión.

Si seleccionamos las pestañas anteriores se abre su formulario asociado.

A continuación se expone un resumen del comportamiento de cada uno de los formularios asociados:

- “Clientes”:

The screenshot shows a software window titled 'Clientes'. It has a left sidebar with menu items: 'Alta / Baja / Modif', 'Crear Cliente', 'Eliminar Cliente', 'Modificar Cliente', 'Salir', and 'Salir'. The main area contains two tables. The top table, 'Rejilla de Clientes', has columns: Id, Fecha alta, CIF, Nombre, Dirección, CP, Localidad, and Pro. It lists three clients, with the first one selected. The bottom table, 'Rejilla de Avisos', has columns: Id, Id Cliente, tecnicoId, Fecha recepción, Situación, Fecha realización, and Trabajo a realizar. It lists two service requests, with the first one selected.

Id	Fecha alta	CIF	Nombre	Dirección	CP	Localidad	Pro
3	12/01/2010	33333	Comunidad de Propietarios	C/ Mirlo, 12	28055	Getafe	Ma
2	11/01/2010	22222	Alejandro Martínez Antón	Avda Constitución 47, 4ª A	28051	Leganés	Ma
1	10/01/2010	11111	Carlos Martín García	C/ Fuengirola 1, 1ª A	28001	Madrid	Ma

Id	Id Cliente	tecnicoId	Fecha recepción	Situación	Fecha realización	Trabajo a realizar
7	3	1	05/01/2013	PENDIENTE		No sale agua caliente en
5	3	1	05/01/2013	REALIZADO	07/01/2013	No sale agua caliente en ni

**Figura 9. Aplicación cliente de gestión: Ventana de Clientes.**

El formulario “Clientes” es una ventana maestro/detalle donde los datos de la rejilla inferior de avisos (detalles) dependen del ítem seleccionado en la rejilla superior de clientes (maestro).

Ambas rejillas incorporan una barra lateral izquierda que muestra las opciones disponibles: alta, eliminación y modificación.

La elección de cualquiera de las tres supone la apertura de un nuevo formulario de campos editables que presenta un aspecto determinado según la opción escogida.

A continuación se ilustra el ejemplo de un alta de un aviso para el cliente seleccionado (3, Comunidad de Propietarios):

The screenshot shows a software window titled 'Gestión de Trabajos en Movilidad. Alba Calvo Ruiz'. It has a top menu bar with 'Clientes', 'Empleados', 'Rutas de técnicos', 'Usuarios', 'Mis Datos', and 'Logout'. The left sidebar has menu items: 'Alta / Baja / Modif', 'Crear Cliente', 'Eliminar Cliente', 'Modificar Cliente', 'Salir', and 'Salir'. The main area shows the 'Rejilla de Clientes' table with the first client selected. Overlaid on this is a 'Ficha de aviso' form. The form has fields for 'Id', 'Cli Id' (set to 3), 'Tec Id' (set to 1), 'Técnico asociado' (Pepe Pérez Pérez), 'Trabajo a Realizar' (Reparación del radiador), and 'Situación' (PENDIENTE). There are 'Grabar' and 'Salir' buttons at the bottom.

Id:  Cli Id: 3 Tec Id: 1

Técnico asociado: Pepe Pérez Pérez

Trabajo a Realizar: Reparación del radiador

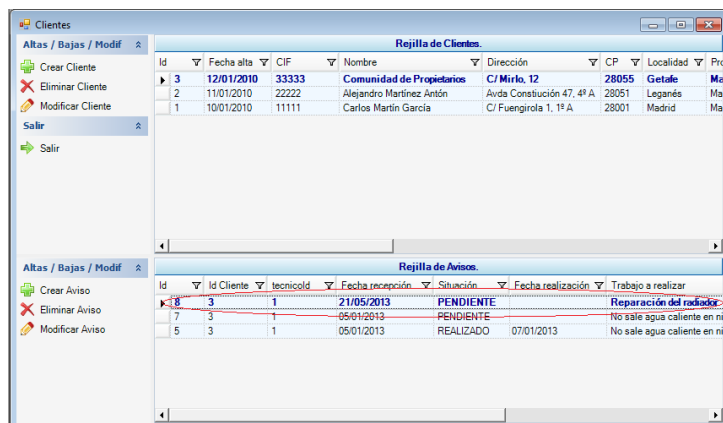
Situación: PENDIENTE

Grabar Salir

**Figura 10. Aplicación cliente de gestión: Alta de aviso.**

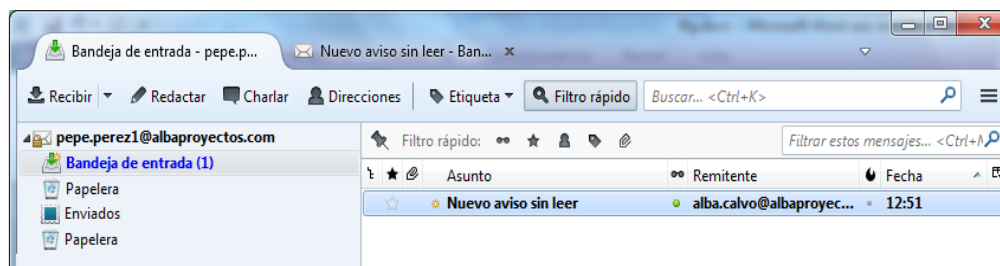
La ficha consta de una serie de campos a rellenar. Al finalizar es necesario elegir la opción “Grabar” si se desea guardar los cambios de forma permanente.

Al grabar el formulario, el alta se ve reflejada en la rejilla de avisos:

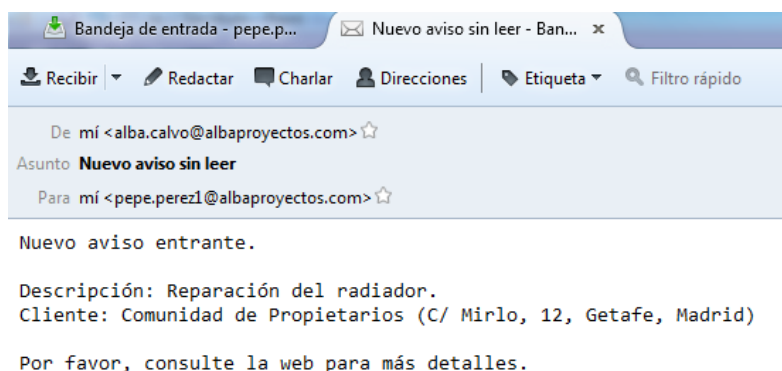


**Figura 11. Aplicación cliente de gestión: Rejilla de avisos con cambios reflejados.**

Cabe destacar que cada vez que se realice el alta de un aviso asociado a un técnico, éste recibirá una notificación en su correo de la empresa:



**Figura 12. Aplicación cliente de gestión: Notificación en el correo de la empresa.**

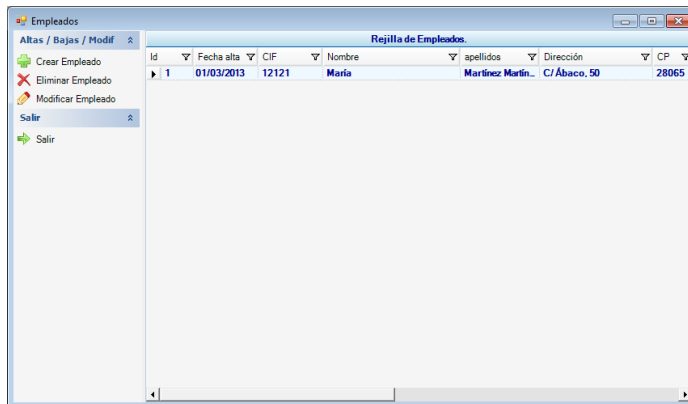


**Figura 13. Aplicación cliente de gestión: Mensaje de notificación de recepción de aviso.**

La funcionalidad de modificación y eliminación es muy similar, por lo que no se incluirán imágenes explicativas de las mismas.

- “Empleados” y “Técnicos”:

Estos formularios presentan el siguiente aspecto:



**Figura 14. Aplicación cliente de gestión: Ventana de Empleados.**

No son ventanas de maestro/detalle, pero su funcionalidad es idéntica a la de la rejilla de avisos del ejemplo anterior, por lo que no se presentarán más diagramas de funcionamiento.

Sin embargo, ambos presentan una característica que los diferencia del resto de formularios de la aplicación: su alta, baja y modificación lleva acciones asociadas.

En ambos formularios, el alta supone la creación de un usuario asociado cuyo nick se crea de la siguiente forma: “e.[nombre].[primer apellido][id empleado]” en el caso de los empleados y “[nombre].[primer apellido][id técnico]” en el caso de los técnicos. Ocurre lo mismo con el email de empleado o técnico pero añadiendo el nombre del dominio “@albaproyectos.com”.

A continuación se ilustra el ejemplo de un alta de empleado:

Id:

CIF:  Fecha Alta:

Nombre:

Apellidos:

Dirección:

Localidad:  Provincia:

CP:

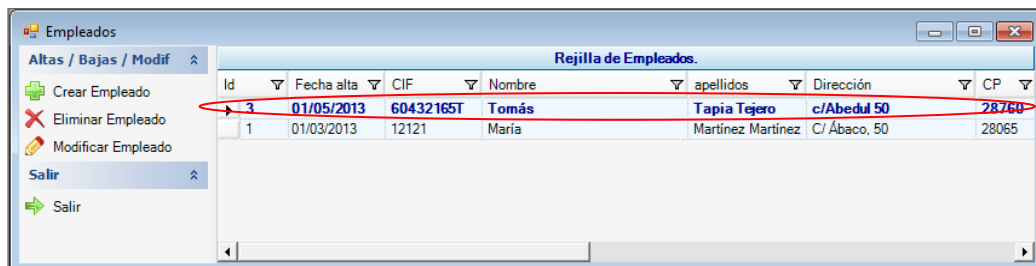
Teléfono:  Móvil:

Email:

Comentarios:

**Figura 15. Aplicación cliente de gestión: Alta de empleado.**

Al igual que en el alta de avisos, se presenta una ficha con campos a rellenar y, tras grabarla, los cambios se ven reflejados en la rejilla de empleados:



**Figura 16. Aplicación de gestión: Rejilla de empleados con cambios reflejados.**

Pero además se crea un usuario asociado, como se puede observar en el formulario “Usuarios”.



**Figura 17. Aplicación cliente de gestión: Rejilla de usuario con cambios reflejados.**

La funcionalidad de modificación y eliminación de un empleado es muy similar y supone la modificación o eliminación del usuario asociado.

- “Tracking de técnicos”:

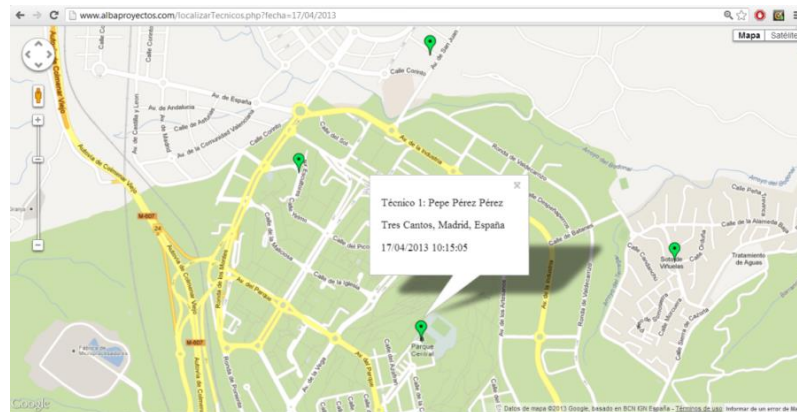


**Figura 18. Aplicación cliente de gestión: Ventana de tracking de técnicos.**

Este formulario cuenta con dos opciones:

- Ubicación de técnicos:

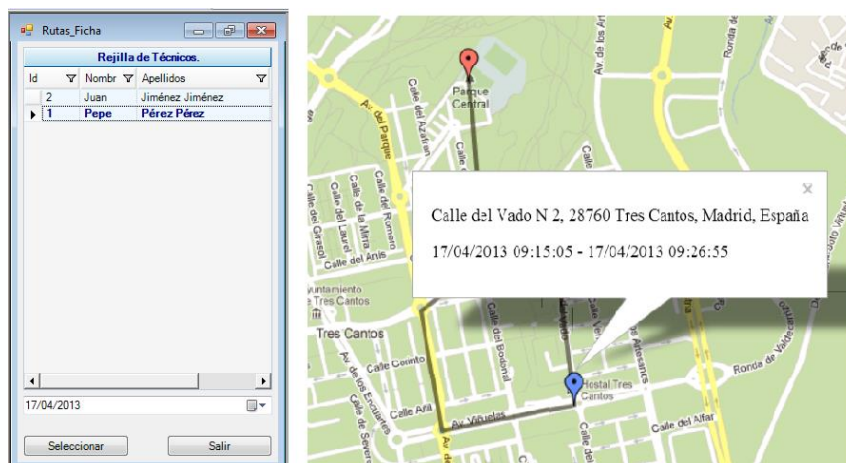
Esta opción se encarga de abrir una instancia del navegador predeterminado y mostrar un mapa con la última posición conocida de todos los técnicos para la fecha actual mostrando la posición de cada técnico con un marcador verde.



**Figura 19. Aplicación cliente de gestión: ubicación de técnicos.**

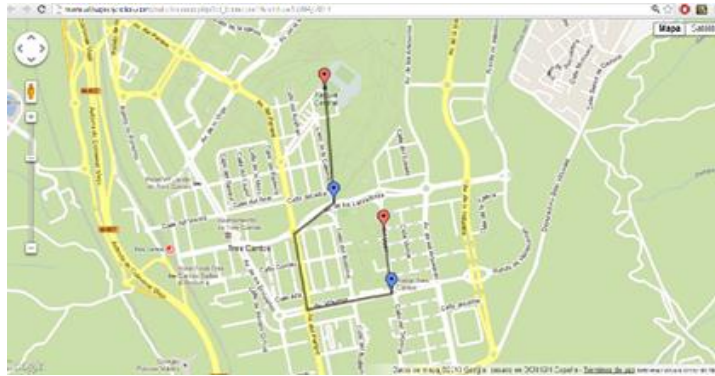
- Ruta de técnicos:

Esta opción se encarga de mostrar un formulario que solicita un técnico y una fecha y, si hay coordenadas de geolocalización disponibles, abre una instancia del navegador predeterminado y muestra un mapa del recorrido del técnico en esa fecha, representando el inicio y el fin del recorrido con un marcador rojo y las paradas del técnico (más de 5 minutos en un punto) en azul. Si se selecciona el marcador se muestra la dirección y la hora de inicio y fin de la parada.



**Figura 20. Aplicación cliente de gestión: Ruta de técnicos 1.**





**Figura 21. Aplicación cliente de gestión: Ruta de técnicos 2.**

- “Usuarios”:

Como se puede observar en la figura 17, la única acción posible de este formulario es la modificación del usuario, concretamente del campo contraseña.

**Figura 22. Aplicación cliente de gestión: Modificación de usuario.**

- “Mis datos”:

Este formulario es común a administrador y empleado (en el caso de empleado los campos nombre y apellidos se encuentran inhabilitados).

Esta ventana permite cambiar el nombre y apellidos del administrador en caso de que se relegue esta tarea en otra persona. También permite cambiar la contraseña.

**Figura 23. Aplicación cliente de gestión: Modificación de datos de usuario.**

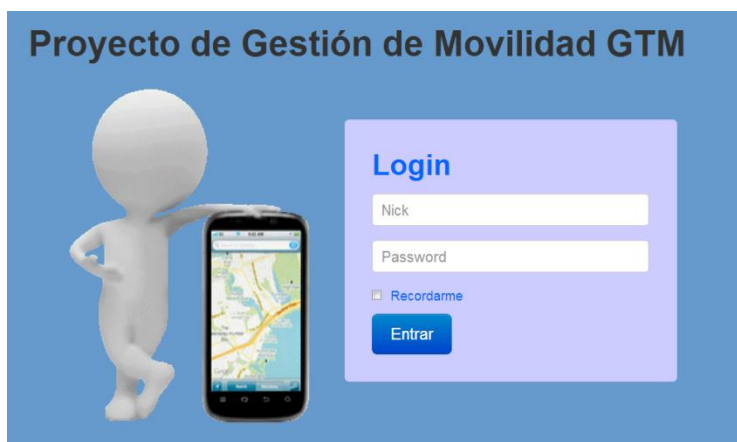
- “Logout”: La selección de esta opción finaliza la sesión.

### 3.2.3 Aplicación Web

El proyecto consta de una Web que contempla dos roles de usuario:

- Administrador: podrá consultar ubicación y rutas de técnicos con información de la posición y duración de cada parada.
- Técnico: tras identificarse, los técnicos podrán visualizar la información de los avisos entrantes (cliente, dirección, trabajo a realizar...) y gestionarlos una vez los hayan marcado como leídos. Los técnicos también podrán consultar un historial de los avisos realizados en el último mes.

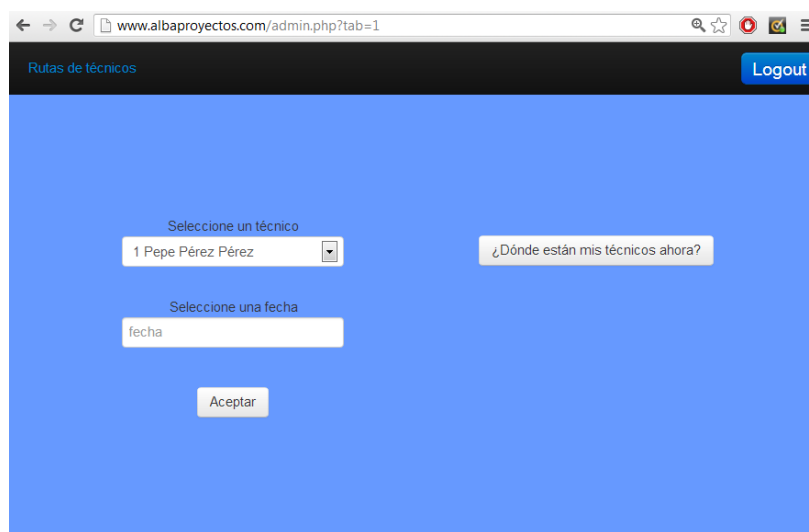
A continuación se expone el funcionamiento de la Web:



**Figura 24. Aplicación Web: Login.**

Inicialmente se muestra la pantalla de login. Si los datos no son correctos se muestra un mensaje de error. Si lo son, se distinguen dos posibles flujos según el rol del usuario.

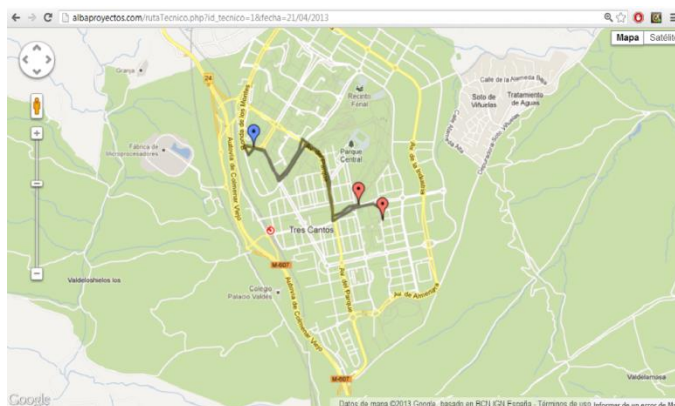
- Rol de administrador:



**Figura 25. Aplicación Web: Menú de administrador.**

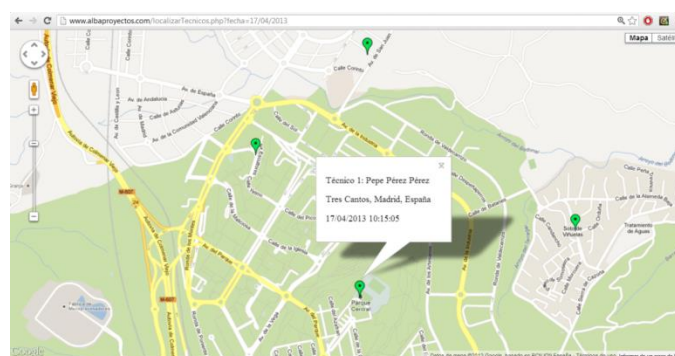
El administrador dispone de dos opciones:

- Obtención de una ruta de técnico seleccionando un técnico y una fecha:



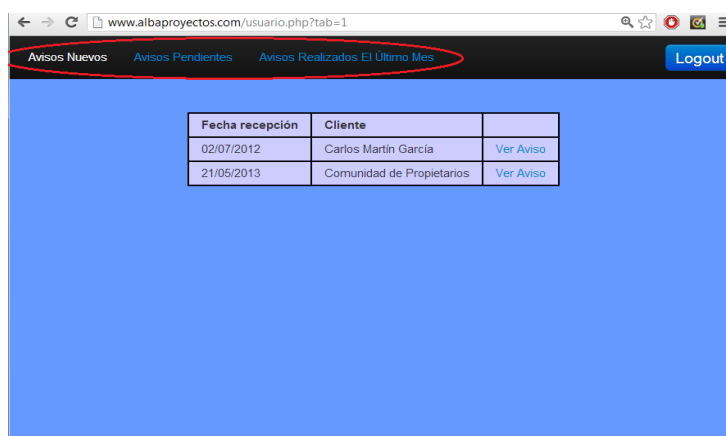
**Figura 26. Aplicación Web: Ruta de técnico.**

- Obtención de la ubicación de todos los técnicos de la empresa en este momento:



**Figura 27. Aplicación Web: Ubicación de técnicos.**

- Rol de técnico:



**Figura 28. Aplicación Web: Menú de técnico.**

La página del técnico consta de tres pestañas:

- Avisos Nuevos



Fecha recepción	Cliente	
02/07/2012	Carlos Martín García	<a href="#">Ver Aviso</a>
21/05/2013	Comunidad de Propietarios	<a href="#">Ver Aviso</a>

**Figura 29. Aplicación Web: Avisos nuevos.**

En esta pestaña figuran los avisos que acaban de ser dados de alta y notificados al técnico por correo. Los avisos deben ser visualizados por el técnico antes de poder ser gestionados. Para ello es necesaria la selección de la opción “Ver Aviso” para ver los detalles y marcarlo como visto:



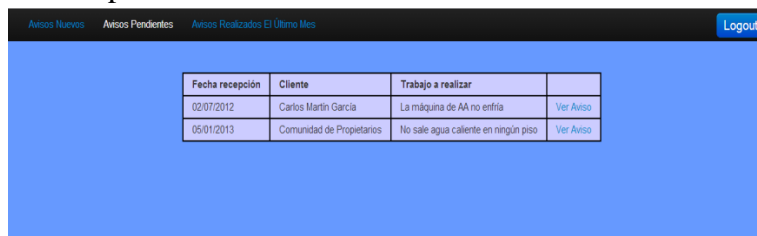
DATOS DEL AVISO	
Id aviso	6
Cliente	Carlos Martín García
Dirección	C/ Fuengirola 1, 1ª A Madrid, Madrid (28001)
Teléfono	91 1111111 91 1222222 611 1111111
Fecha recepción	02/07/2012
Situación	PENDIENTE
Trabajo a realizar	La máquina de AA no enfría

[Aceptar](#)

**Figura 30. Aplicación Web: Datos de aviso.**

Una vez visualizado, el aviso pasará a estado pendiente y el técnico podrá proceder a su gestión.

- Avisos pendientes



Fecha recepción	Cliente	Trabajo a realizar	
02/07/2012	Carlos Martín García	La máquina de AA no enfría	<a href="#">Ver Aviso</a>
05/01/2013	Comunidad de Propietarios	No sale agua caliente en ningún piso	<a href="#">Ver Aviso</a>

**Figura 31. Aplicación Web: Avisos pendientes.**

Esta pestaña muestra los avisos marcados como pendientes que aún no han sido realizados.

El técnico puede consultar los detalles de los mismos al igual que en el caso anterior pero ahora dispone de la opción “Anular/Finalizar” para cumplimentar y terminar el aviso con estado finalizado o anulado.

DATOS DEL AVISO	
Id aviso	7
Cliente	Comunidad de Propietarios
Dirección	C/ Mirlo, 12 Getafe, Madrid (28055)
Teléfono	91 3111111 91 3222222 633 111111
Fecha recepción	05/01/2013
Situación	PENDIENTE
Trabajo a realizar	No sale agua caliente en ningún piso

Volver Anular / Finalizar

**Figura 32. Aplicación Web: Datos de aviso con opción de cumplimentación.**

DATOS DEL AVISO	
Id aviso	7
Cliente	Comunidad de Propietarios
Dirección	C/ Mirlo, 12 Getafe, Madrid (28055)
Teléfono	91 3111111 91 3222222 633 111111
Fecha recepción	05/01/2013
Situación	PENDIENTE
Trabajo a realizar	No sale agua caliente en ningún piso

Trabajo realizado:

Reparación del calentador de agua.

Situación REALIZADO

Aceptar Cancelar

**Figura 33. Aplicación Web: Cumplimentación de aviso.**

Tras marcar el aviso como realizado o anulado, este pasa al historial de avisos.

- Avisos realizados el último mes

Fecha finalización	Cliente	Trabajo realizado	
08/05/2013	Comunidad de Propietarios	Reparación del calentador de agua.	<a href="#">Ver Aviso</a>

**Figura 34. Aplicación Web: Avisos realizados el último mes.**

En esta pestaña podemos ver los avisos realizados este mes. (El aviso que hemos marcado en el paso anterior como realizado aparece en la tabla).

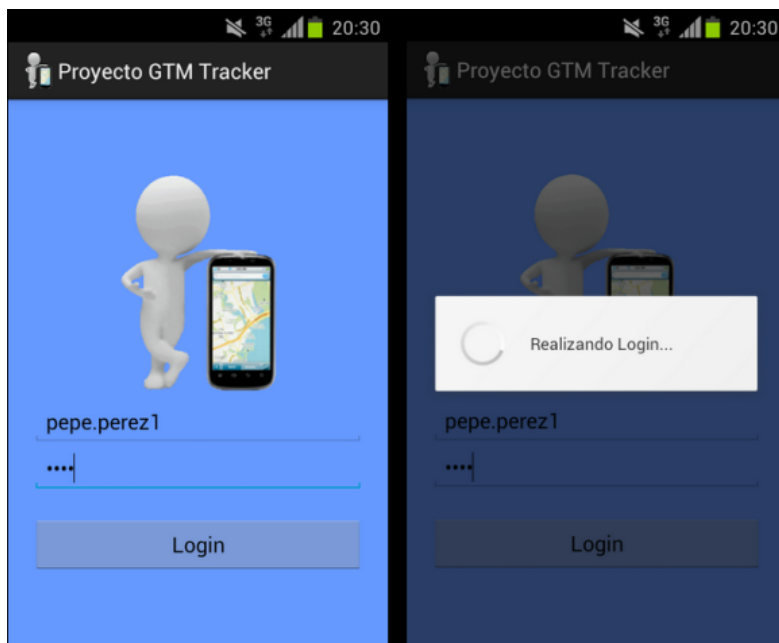
### 3.2.4 Aplicación de geolocalización

El proyecto incluye una aplicación de geolocalización para dispositivos móviles con sistema operativo Android.

Los técnicos deben iniciar la aplicación al comienzo de su turno de trabajo para enviar su posición cada minuto o cada vez que recorran 100 metros y mantenerla activa durante el transcurso de su actividad.

Esta aplicación activa un servicio de localización que sigue activo aunque la aplicación principal pase a segundo plano o incluso sea cerrada por el planificador de procesos, por lo que el técnico podrá hacer un uso normal de su teléfono.

A continuación se expone el diagrama de flujo de la aplicación:

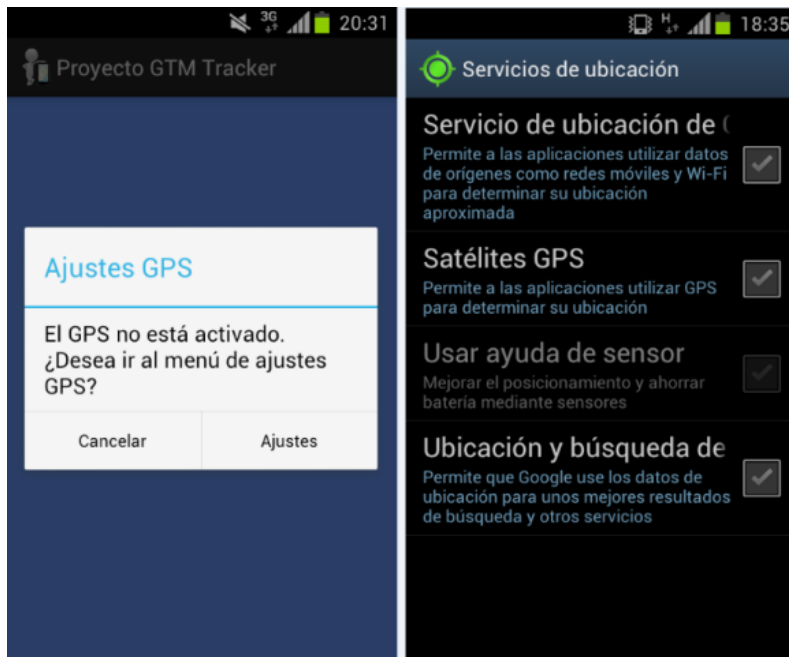


**Figura 35. Aplicación Android: Login.**

Inicialmente se muestra la pantalla de login donde el técnico debe autenticarse.

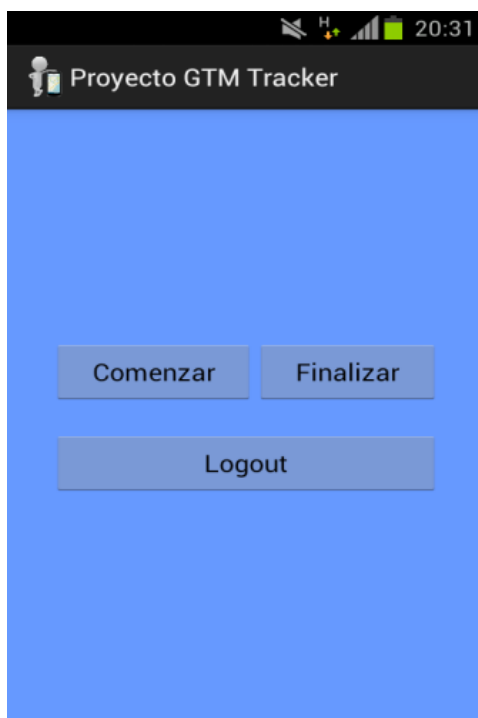
Si la autenticación es correcta se accede al menú principal de la aplicación.

La aplicación detecta la habilitación del GPS y, si se halla inhabilitado, muestra al usuario un mensaje antes de mostrar el menú. Si el usuario selecciona la opción “Ajustes” se accede a los ajustes de ubicación de Android.



**Figura 36. Aplicación Android: Habilitación de GPS.**

Independientemente de la elección del usuario de habilitar o no el GPS, a continuación se muestra el menú:



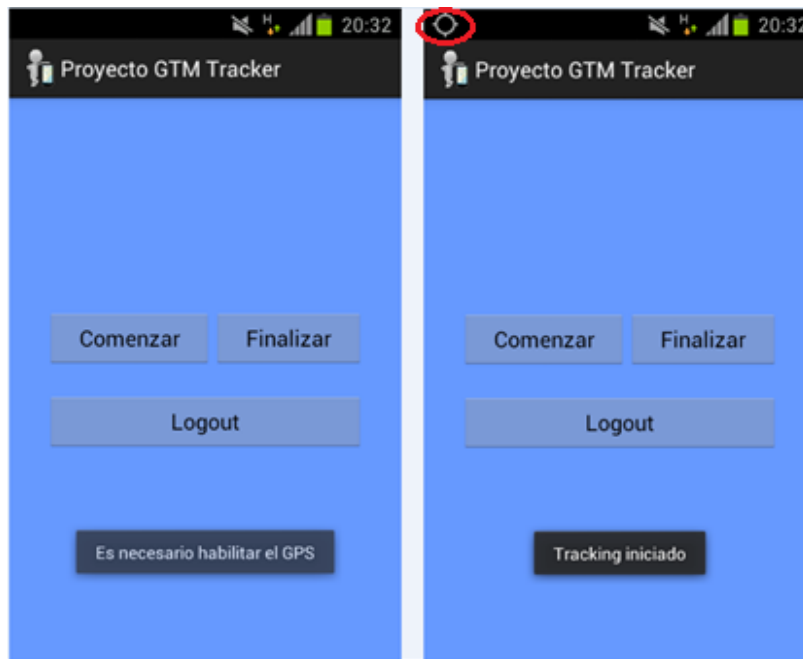
**Figura 37. Aplicación Android: Menú.**

La interfaz del menú es muy sencilla y consta de tres botones:

- Comenzar: inicia un servicio de geolocalización en un hilo diferente al de la aplicación principal.
- Finalizar: detiene el servicio de geolocalización.
- Logout: detiene el servicio en caso de que esté activo y cierra la sesión de usuario.  
(Para finalizar la aplicación no es necesario hacer Logout, sino que basta con pulsar la tecla back del teléfono. Esta forma de salir de la aplicación es recomendable, ya que así el usuario no tiene que autenticarse cada vez que utiliza la aplicación).

Cabe destacar que al pulsar el botón “Comenzar”, el servicio sólo comenzará si el GPS está habilitado. De lo contrario se notificará al usuario que debe activar el GPS.

Si todo va bien, aparecerá en la barra superior un icono parpadeante que indica que se está localizando la posición del usuario vía GPS, que debe desaparecer al finalizar el servicio.



**Figura 38. Aplicación Android: Servicio de tracking.**



## 4 Desarrollo

---

A continuación se analizará en detalle la estructura y el comportamiento de cada una de las partes del proyecto GTM desde el punto de vista del desarrollador.

### 4.1 Base de datos

#### 4.1.1 Tecnologías empleadas

Para realizar la configuración, gestión y administración de la base de datos se ha utilizado la herramienta SQL Server Management Studio 2012, un sistema basado en el modelo relacional con soporte para lenguajes ANSI SQL.

Esta herramienta permite la creación la base de datos en un servidor SQL Server remoto y posibilita la gestión de la misma.

### 4.2 Aplicación cliente de gestión

#### 4.2.1 Tecnologías empleadas

Para codificar la aplicación cliente se ha empleado el entorno de desarrollo Microsoft Visual Studio 2010 empleando el lenguaje de programación C# y el framework .NET.

Para el desarrollo de la interfaz gráfica se ha empleado Infragistics, una herramienta de desarrollo de interfaces de usuario para ASP.NET que incluye controles para su uso dentro de formularios.

#### 4.2.2 Estructura

La aplicación consta de varias capas:

- Modelo de datos (ModeloDTO)
- Interfaz (MovilidadPresentacion)
- Capa de negocio (Negocio)

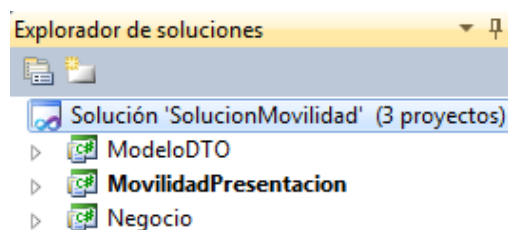


Figura 39: Capas de la aplicación cliente.

#### 4.2.2.1 Negocio y ModeloDTO

Estas dos capas están estrechamente relacionadas entre sí y se encargan del acceso a la información de la base de datos.

Para acceder a esta información se utiliza el Entity Framework de ADO.NET, que consiste en un mapeo objeto-relacional (ORM) para el framework .NET que permite crear aplicaciones con acceso a datos programando contra un modelo conceptual de aplicación en lugar de directamente contra un esquema de almacenamiento relacional. Se consigue así disminuir la cantidad de código y simplificar el mantenimiento de la aplicación.

Para realizar el mapeo es necesario generar un modelo a partir de la base de datos en nuestra aplicación. Visual Studio dispone de un asistente que crea clases para todas las tablas que representan entidades y crea una clase contexto (ProyectoMovilidadContext) que controla la conexión a la base de datos y el flujo de datos.

En la capa de negocio se encuentran los DAOs (Data Access Object), objetos que permiten el acceso a la base de datos de forma abstracta a través del conjunto de métodos que proporcionan. Desde la aplicación se puede acceder al modelo utilizando el patrón de diseño que ofrece el DAO de forma transparente.

La capa de modelo de datos incluye los DTOs (Data Transfer Object), objetos que sirven para transportar datos.

El DAO, tras una petición del controlador, rellena un DTO con los datos deseados y lo devuelve al controlador.

El uso de DAOs y DTOs permite así la separación entre el controlador y el modelo de datos y aporta al código gran flexibilidad, pues ante cualquier cambio en la forma de acceder a los datos (cambio de base de datos, obtención de datos de servicios Web, etc) el modelo no se ve afectado, simplemente sería necesario cambiar los DAOs involucrados.

A continuación se expone un ejemplo de DAO y DTO de la tabla cliente para ilustrar la explicación anterior:

DAO cliente:

```
namespace Negocio.Servicio
{
    public class ClienteServicio : BaseService
    {
        public ModeloDTO.Cliente.ClienteFichaDto getClienteFichaDto(Int32 idCliente)
        {
            var contexto = new ProyectoMovilidadContext();
            var listaClientes = from cli in contexto.cliente
                               where cli.id == idCliente
                               select new ModeloDTO.Cliente.ClienteFichaDto() {
                                   id = cli.id,
                                   fechaAlta = cli.fecha_alta,
                                   cif = cli.cif,
                                   nombre = cli.nombre,
                                   direccion = cli.direccion,
                                   d_postal = cli.d_postal,
                                   localidad = cli.localidad,
                                   provincia = cli.provincia,
                                   tlf = cli.tlf,
            };
            return listaClientes.FirstOrDefault();
        }
    }
}
```

```

        tlf_2 = cli.tlf_2,
        tlf_movil = cli.tlf_movil,
        fax = cli.fax,
        email = cli.email,
        Web = cli.Web,
        comentarios = cli.comentarios
    };
    ModeloDTO.Cliente.ClienteFichaDto dtoFichaCli = listaClientes.First();
    contexto.Dispose();
    return dtoFichaCli;
}

public List<ModeloDTO.Cliente.ClienteFichaDto> getListaClienteFichaDto() {
    var contexto = new ProyectoMovilidadContext();
    var listaClientes = from cli in contexto.cliente
    select new ModeloDTO.Cliente.ClienteFichaDto() {
        id = cli.id,
        fechaAlta = cli.fecha_alta,
        cif = cli.cif,
        nombre = cli.nombre,
        direccion = cli.direccion,
        d_postal = cli.d_postal,
        localidad = cli.localidad,
        provincia = cli.provincia,
        tlf = cli.tlf,
        tlf_2 = cli.tlf_2,
        tlf_movil = cli.tlf_movil,
        fax = cli.fax,
        email = cli.email,
        Web = cli.Web,
        comentarios = cli.comentarios
    };

    List<ModeloDTO.Cliente.ClienteFichaDto> listaDto =
        listaClientes.OrderByDescending(cl => cl.id).ToList();
    contexto.Dispose();
    return listaDto;
}

public void eliminaCliente(Int32 idCliente) {
    var contexto = new ProyectoMovilidadContext();
    cliente clien = (from cli in contexto.cliente
        where cli.id == idCliente
        select cli).First();

    contexto.DeleteObject(clien);
    contexto.SaveChanges();
    contexto.Dispose();
}

public Int32 altaCliente(ModeloDTO.Cliente.ClienteFichaDto dtoCliente) {
    cliente nuevoCli = new cliente() {
        fecha_alta = dtoCliente.fechaAlta,
        cif = dtoCliente.cif,
        nombre = dtoCliente.nombre,
        direccion = dtoCliente.direccion,
        d_postal = dtoCliente.d_postal,
        localidad = dtoCliente.localidad,
        provincia = dtoCliente.provincia,
        tlf = dtoCliente.tlf,
        tlf_2 = dtoCliente.tlf_2,
        tlf_movil = dtoCliente.tlf_movil,
        fax = dtoCliente.fax,
        email = dtoCliente.email,
        Web = dtoCliente.Web,
        comentarios = dtoCliente.comentarios
    };

    var contexto = new ProyectoMovilidadContext();

```

```

        contexto.cliente.AddObject(nuevoCli);
        contexto.SaveChanges();
        contexto.Dispose();

        dtoCliente.id = nuevoCli.id;
        return nuevoCli.id;
    }

    public void modificaCliente(ModeloDTO.Cliente.ClienteFichaDto dtoCliente) {
        var contexto = new ProyectoMovilidadContext();
        cliente clienteModif = (from cli in contexto.cliente
                                where cli.id == dtoCliente.id
                                select cli).First();

        clienteModif.fecha_alta = dtoCliente.fechaAlta;
        clienteModif.cif = dtoCliente.cif;
        clienteModif.nombre = dtoCliente.nombre;
        clienteModif.direccion = dtoCliente.direccion;
        clienteModif.d_postal = dtoCliente.d_postal;
        clienteModif.localidad = dtoCliente.localidad;
        clienteModif.provincia = dtoCliente.provincia;
        clienteModif.tlf = dtoCliente.tlf;
        clienteModif.tlf_2 = dtoCliente.tlf_2;
        clienteModif.tlf_movil = dtoCliente.tlf_movil;
        clienteModif.fax = dtoCliente.fax;
        clienteModif.email = dtoCliente.email;
        clienteModif.Web = dtoCliente.Web;
        clienteModif.comentarios = dtoCliente.comentarios;

        contexto.SaveChanges();
        contexto.Dispose();
    }
}

```

Se puede observar que el DAO proporciona una serie de métodos de obtención de datos en forma de DTOs (ModeloDTO.Cliente.ClienteFichaDto): `getClientefichaDto`, `getListafichaDto`, `eliminarCliente`, `altaCliente` y `modificarCliente`.

Si analizamos cualquiera de ellos en detalle podemos ver que instancian una variable contexto de la clase `ProyectoMovilidadContext`. Esta clase es el modelo que se ha generado a partir del asistente de Visual Studio para controlar la conexión a la base de datos y el flujo de datos.

A continuación se realiza la consulta utilizando LINQ, un componente de la plataforma .NET que permite agregar consultas nativas semejantes a las de SQL a lenguajes de .NET Framework como es C#. Esta consulta devuelve un objeto DTO a cuyos campos podemos acceder desde la aplicación. Este objeto tiene la siguiente forma:

```

namespace ModeloDTO.Cliente
{
    public class ClienteFichaDto
    {
        public ClienteFichaDto()
        {
            fechaAlta = DateTime.Now;
            cif = "";
            nombre = "";
            direccion = "";
            d_postal = "";
            localidad = "";
            provincia = "";
            tlf = "";
            tlf_2 = "";

```

```

        tlf_movil = "";
        fax = "";
        email = "";
        Web = "";
        comentarios = "";
    }

    public Int32 id { get; set; }
    public DateTime? fechaAlta { get; set; }
    public string cif { get; set; }
    public string nombre { get; set; }
    public string direccion { get; set; }
    public string d_postal { get; set; }
    public string localidad { get; set; }
    public string provincia { get; set; }
    public string tlf { get; set; }
    public string tlf_2 { get; set; }
    public string tlf_movil { get; set; }
    public string fax { get; set; }
    public string email { get; set; }
    public string Web { get; set; }
    public string comentarios { get; set; }
}

```

#### 4.2.2.1.1 MovilidadPresentacion

Esta capa representa la lógica e interfaz de la aplicación.

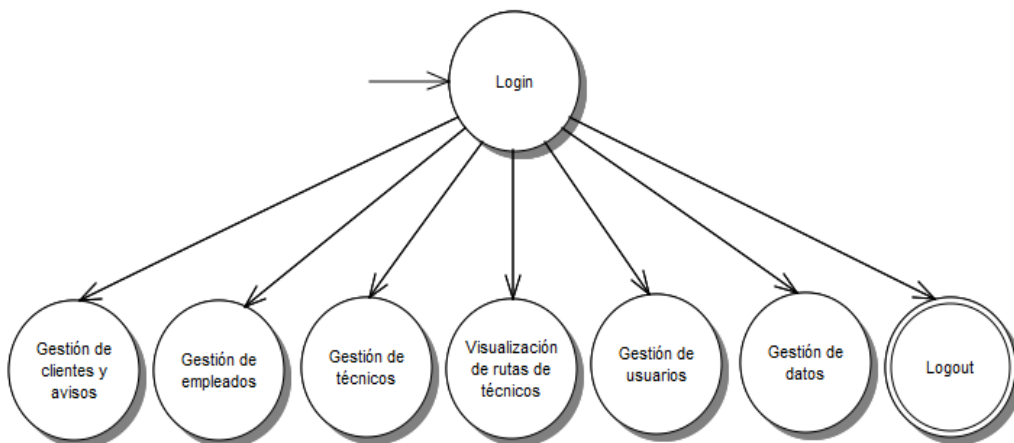
Contiene todos los Windows Forms de la aplicación que llevan una funcionalidad asociada (cuando se muestra el formulario, cuando el usuario interactúa con el formulario, etc).

Es en estos formularios en los que se hace uso de los controles proporcionados por Infragistics, como por ejemplo la rejilla de datos UltraGrid, los botones UltraButton, etc.

### 4.2.3 Diagramas

#### 4.2.3.1 Máquina de estados

##### 4.2.3.1.1 Administrador



**Figura 40. Aplicación cliente de gestión: Máquina de estados de administrador.**

#### 4.2.3.1.2 Empleado

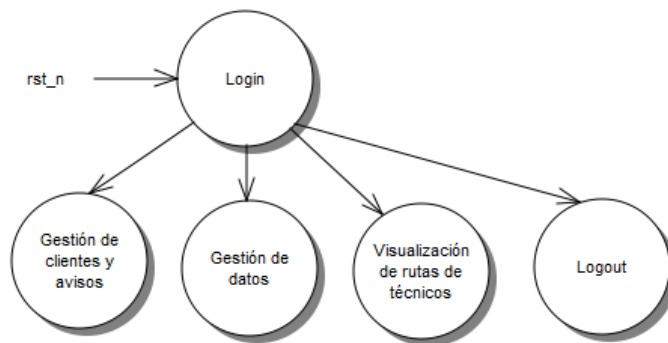


Figura 41. Aplicación cliente de gestión: Máquina de estados de empleado.

#### 4.2.3.2 Diagrama UML

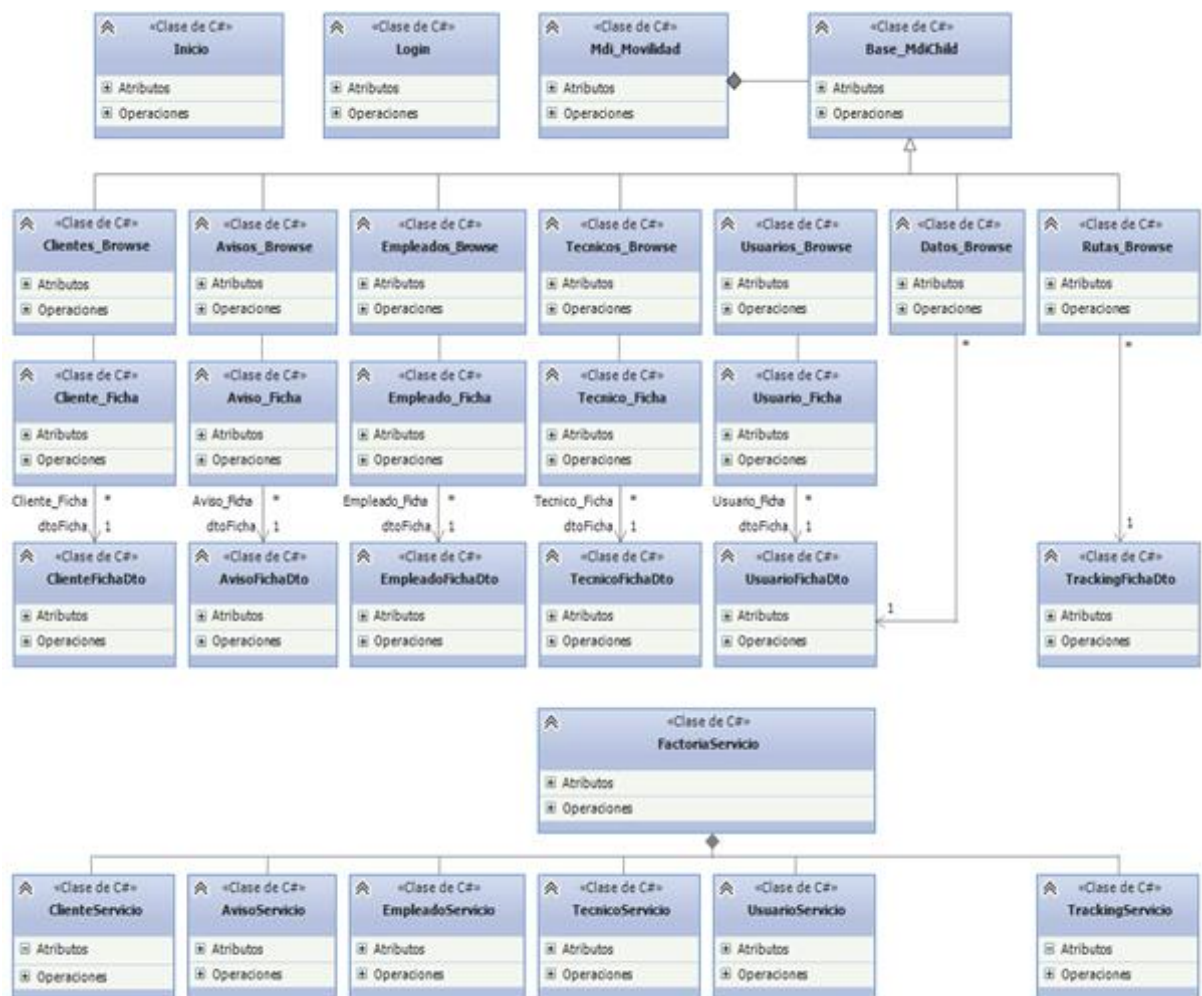
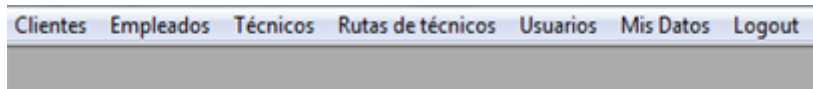


Figura 42. Aplicación cliente de gestión: Diagrama UML.

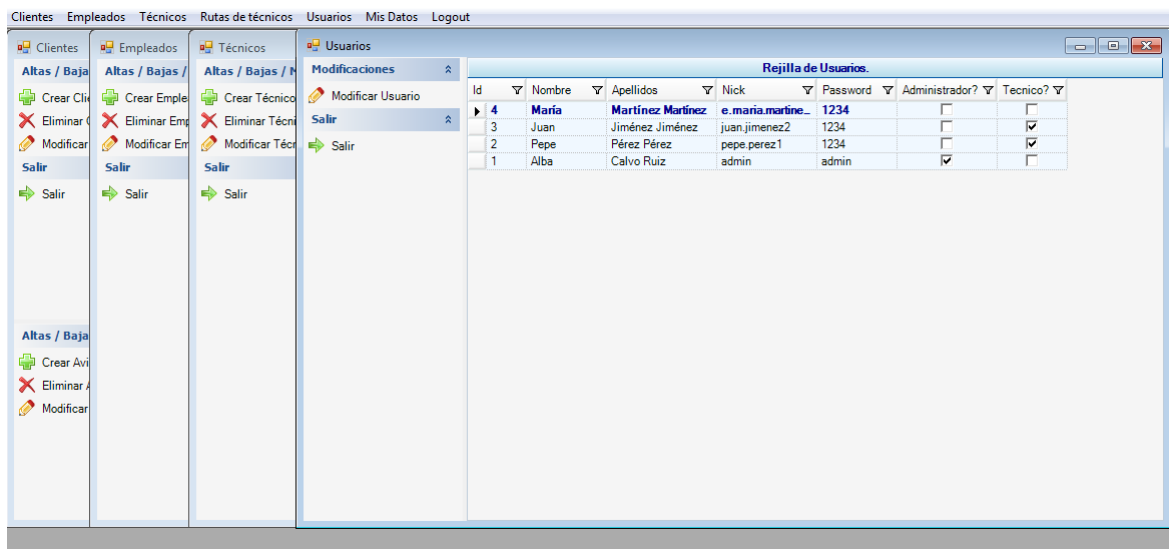
El diagrama de la aplicación abarca varias clases y se explicará brevemente a continuación:

En orden de aparición se presentan los formularios de inicio y login. A continuación se muestra el formulario Mdi\_Movilidad:



**Figura 43. Aplicación cliente de gestión: Formulario MDI 1.**

Este formulario tiene comportamiento MDI, que quiere decir que sus ventanas hijas (MdiChild) residen bajo una única ventana madre. Con esto se consigue que al pulsar en cada una de las pestañas del menú puedan mantenerse a la vez abiertas en el mismo espacio como se muestra en la imagen inferior:



**Figura 44. Aplicación cliente de gestión: Formulario MDI 2.**

Este es el motivo de que todas las clases acabadas en \_Browse que se encargan de presentar el listado de avisos, clientes, empleados... sean del tipo Base\_MdiChild como muestra la relación is\_a del diagrama.

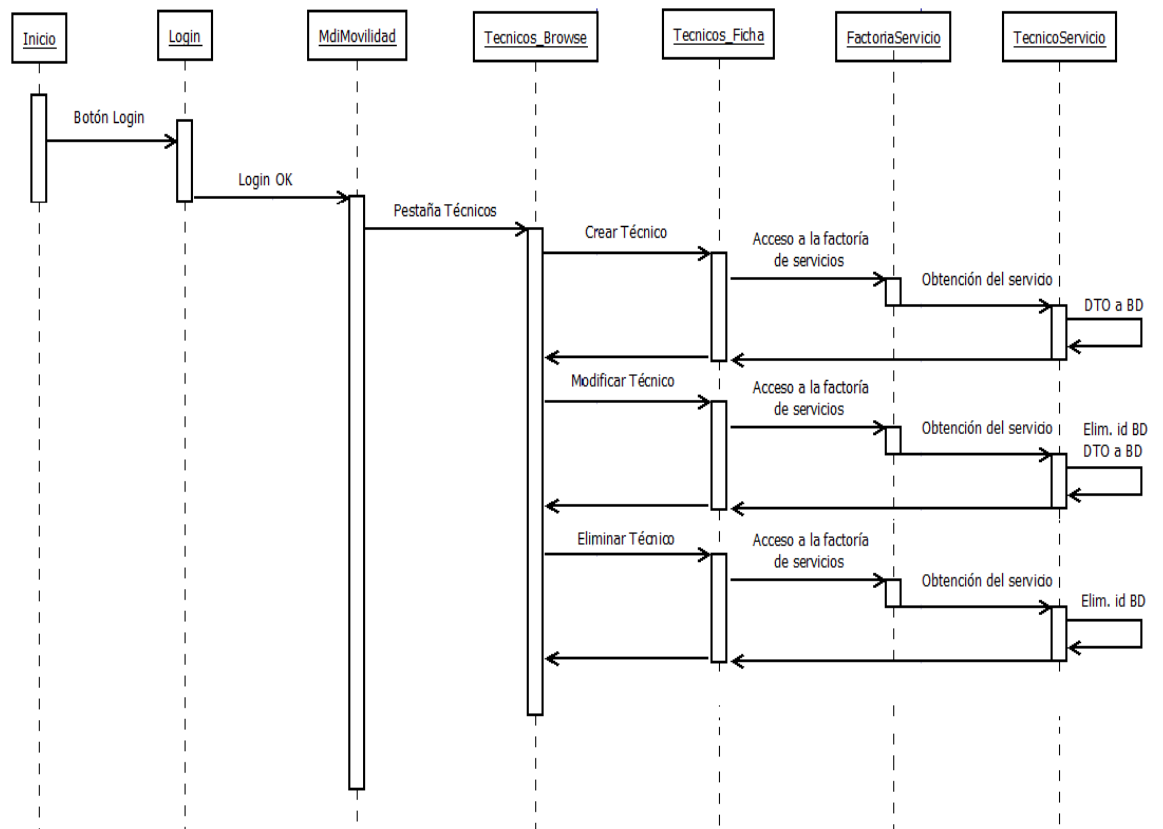
Podemos observar que cada \_Browse tiene una clase \_Ficha asociada. Esta es la ficha de alta, modificación y baja que se presenta al hacer click en las opciones de la barra lateral izquierda de cada MdiChild y que tendrá un aspecto y una habilitación de campos distinta según la opción escogida.

Cada una de estas fichas está relacionada con un objeto de transmisión de datos DTO que ejerce de intermediario entre el formulario y el servicio que ejerce el papel de objeto de acceso a datos DAO y que permite la relación entre la aplicación y la base de datos. Así, en el formulario se accederá al servicio correspondiente a la ficha (ej: ClienteServicio si nos encontramos en Cliente\_Ficha), que se encargará de gestionar la correspondencia entre los

campos del DTO y los de la ficha y de realizar las modificaciones pertinentes en la base de datos.

El acceso a los servicios se realizará a través de la clase FactoriaServicio, una factoría que evita la replicación de los mismos. Esta factoría se encarga de la creación de clases XServicio sólo si no han sido instanciadas previamente o de la devolución de la instancia existente en caso contrario.

#### 4.2.3.3 Diagrama de secuencia



**Figura 45. Aplicación cliente de gestión: Diagrama de secuencia.**

Este es el diagrama de secuencia de una de las posibles opciones: la autenticación y la gestión de técnicos. Es muy representativo y el resto de opciones presenta un flujo muy similar, por lo que no se incluye un diagrama para cada opción.



## 4.3 Aplicación Web

### 4.3.1 Tecnologías empleadas

Para realizar la aplicación Web se ha utilizado PHP5 y jQuery, una librería de Javascript que facilita una API fácil de usar que funciona en multitud de navegadores y que ha simplificado la tarea de desarrollo.

También se ha empleado la API de Google Maps para poder incluir mapas en la aplicación Web. Esta API es libre aunque con un límite de solicitudes diarias que son suficientes para el proyecto y además está muy bien documentada.

Para organizar el proyecto de forma sencilla se ha empleado el entorno de desarrollo Eclipse.

### 4.3.2 Estructura

La estructura de la Web se resume en el siguiente esquema:

```
directorio raíz (/)
+ css (/css)
+ imagenes (/imagenes)
+ javascript (/javascript)
  - jquery.js
admin.php
aviso.php
aviso_historial.php
aviso_nuevos.php
aviso_pendientes.php
checkNickPass.php
Conexion.class.php
index.php
localizarTecnicos.php
login.php
logout.php
rutaTecnico.php
servicioInsertarGeo.php
servicioLogin.php
usuario.php
```

- css: esta carpeta incluye las hojas de estilo de la Web.
- imagenes: esta carpeta incluye todas las imágenes utilizadas en la página Web.
- javascript: esta carpeta incluye el archivo de la librería jQuery.
- Ficheros PHP siguiendo el orden de aparición:
  - index.php: se encarga de redirigir al usuario a login.php

- login.php: página de autenticación. Tras rellenar el formulario con nick, password y opción “Recordarme”, si los campos están rellenos se redirige a checkNickPass.php y se comprueba que el usuario existe. De lo contrario muestra el mensaje “Rellene todos los campos”.
- checkNickPass.php: comprueba los datos de autenticación. Si son correctos redirige a admin.php o a usuario.php según el rol. De lo contrario vuelve a login.php y muestra en rojo el mensaje “Nick o contraseña incorrectos”.
- admin.php (rol administrador): dispone de una única pestaña “Ruta de Técnicos” que presenta dos opciones: presentación de la ruta de un técnico dada una fecha (rutaTecnico.php) y presentación de la ubicación de los técnicos en este momento (localizarTecnicos.php).
- rutaTecnico.php: muestra el mapa de la ruta de un técnico dado su nombre y una fecha.
- localizarTecnicos.php: muestra la última localización conocida de los técnicos en la fecha actual.
- usuario.php (rol de técnico): dispone de tres pestañas que muestran en una tabla los avisos nuevos, pendientes y el historial de avisos. Si visualizamos un ítem de la tabla se redirige a aviso.php, que muestra los datos de un aviso.
- logout.php: finaliza la sesión. Si el usuario había seleccionado la opción recordarme al iniciar sesión se destruye la cookie correspondiente para que la próxima vez sea necesaria su identificación.
- servicioInsertarGeo.php y servicioLogin.php: son servicios PHP para insertar coordenadas en la base de datos y para realizar el login desde la aplicación móvil. Se explicarán más adelante en el punto destinado a la aplicación Android.
- Conexion.class: varios de los ficheros anteriores precisan el acceso a la base de datos (login, avisos, rutas). Para simplificar el código y hacerlo más legible se ha creado una clase Conexion que proporciona una serie de métodos de interacción con la base de datos: conectar, ejecutar, obtener\_resultado, obtener\_fila, getCountResult.

```
<?php
/* Clase encargada de gestionar las conexiones a la base de datos */
Class Conexion {
    private $dsn = "Driver={SQL Server};Server=███;Database=
        ███;Integrated
        Security=SSPI;Persist Security Info=False;";
    private $conexion;
    private $result;
    private $array;
    static $_instance;

    /* Función para evitar que el objeto pueda ser creado mediante new*/
    private function __construct(){
```

```

        $this->conectar();
    }

    /* Patrón Singleton. Evita la clonación del objeto */
    private function __clone(){ }

    /*Función encargada de crear, si es necesario, el objeto */
    public static function getInstance(){
        if (!(self::$_instance instanceof self))
            self::$_instance=new self();
        return self::$_instance;
    }

    /* Realiza la conexión a la base de datos */
    private function conectar(){
        $this->conexion = odbc_connect($this->dsn, ■■■, ■■■);
        if (!$this->conexion)
            exit("<strong>Error al intentar conectar con el origen de datos
            ODBC.</strong>");
    }

    /* Método para ejecutar una sentencia SQL */
    public function ejecutar($sql){
        $this->result = odbc_exec($this->conexion,$sql)or die(exit("Error
en
        odbc_exec"));
        return $this->result;
    }

    /* Método para obtener el resultado de la sentencia SQL */
    public function obtener_resultado($result){
        $this->array=odbc_result($result, 1);
        return $this->array;
    }

    /* Método para obtener una fila de resultados de la sentencia SQL */
    public function obtener_fila($result,$fila){
        $this->array=odbc_fetch_array($result,$fila);
        return $this->array;
    }

    /* Método para obtener el número de filas de la sentencia SQL ejecutada */
    public function getCountResult() {
        return odbc_num_rows($this->result);
    }
    }
?>

```

Para utilizar esta clase en los ficheros PHP es necesario incluir la sentencia “require 'Conexion.class.php';” e instanciar una variable de conexión: \$conexion = Conexion::getInstance();

Tras realizar esto ya se puede llamar a los métodos de la clase para ejecutar consultas (\$conexion->ejecutar(“SELECT \* FROM usuario;”).

### 4.3.3 Diagramas

#### 4.3.3.1 Máquina de estados

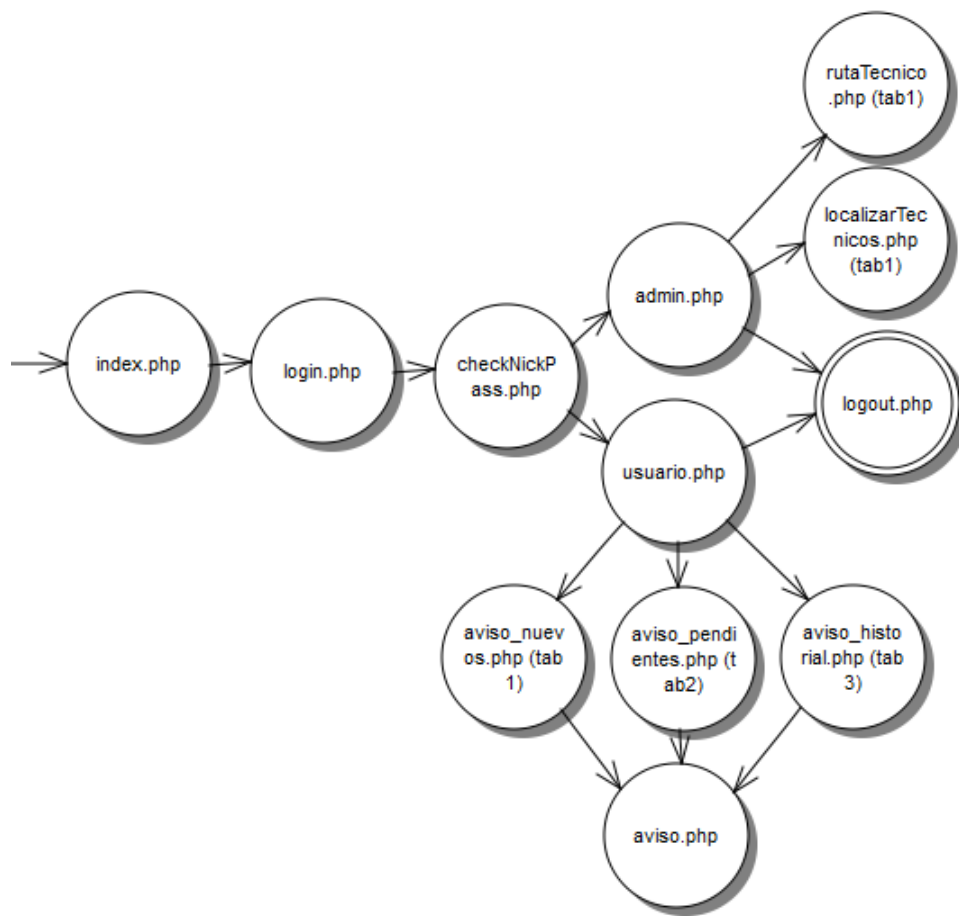


Figura 46. Aplicación Web: Máquina de estados.

## 4.4 Aplicación de geolocalización

### 4.4.1 Tecnologías empleadas

La aplicación móvil se ha desarrollado en Android debido a su posición dominante en el mercado de los dispositivos móviles.

Para ello se ha trabajado con el entorno Eclipse para desarrollo de aplicaciones Android.

### 4.4.2 Estructura

La aplicación Android de geolocalización consta de cuatro clases:

- Login.java
- TareaLogin.java
- TrackerActivity.java
- MyService.java

Inicialmente se muestra la ventana asociada a la actividad Login (Login.java), donde el técnico introduce su nick y password.

A continuación, se comprueba que los datos son correctos. Esta comprobación se realiza mediante la llamada a una tarea asíncrona (TareaLogin) que se ejecuta en otro hilo y así la interfaz de la aplicación no se ve ralentizada dando impresión de mal funcionamiento. Esta tarea muestra un diálogo de proceso mientras comprueba la corrección del usuario y contraseña. Para ello se comunica con el servicio servicioLogin.php enviando los parámetros nick y password por método POST y recibiendo un fichero en formato JSON (formato ligero de intercambio de datos) del servidor como respuesta. La tarea parsea entonces la respuesta JSON que, si es distinta de “-1”, indica que el login se ha realizado con éxito.

```
String url="http://www.albaproyectos.com/servicioLogin.php";

// Envío post y recepcion de JSON
String json = getJSONFromUrl(url, username, password);
logged=false;
if(!json.equals("-1"))
    logged=true;
```

Si el login es incorrecto, se mantiene al usuario en la actividad Login mostrando un mensaje de error. Si la autenticación es correcta, se guarda el usuario en el fichero de preferencias de Android y se inicia la actividad TrackerActivity.

```
SharedPreferences settings = PreferenceManager.getDefaultSharedPreferences(contexto);
SharedPreferences.Editor editor = settings.edit();
editor.putString("id_usu", aux);
editor.commit();
```

En la actividad TrackerActivity el técnico tiene la opción de iniciar y finalizar la geolocalización. Esta actividad se encarga de iniciar o finalizar un servicio (MyService) que envía las coordenadas del técnico cada cien metros o cada minuto. Este servicio se

ejecuta en un hilo diferente al de la aplicación principal, por lo que si ésta pasa a segundo plano o es finalizada por el planificador de procesos de Android, el servicio seguirá activo. Para utilizar este servicio basta con declarar una variable de tipo Intent para la clase MyService y llamar al método startService para iniciar el servicio o stopService para finalizarlo:

```
Intent intent = new Intent(this, MyService.class);
startService(intent);
stopService(intent);
```

La clase MyService debe extender Service y sobrescribir ciertos métodos que todo servicio debe tener (onBind, onStartCommand).

A grandes rasgos su funcionamiento es como sigue:

Primero se inicializa una variable de tipo LocationManager que da acceso a los servicios de localización del sistema.

```
LocationManager locationManager;
locationManager = (LocationManager) getSystemService (Context.LOCATION_SERVICE);
```

Los servicios de localización del sistema permiten obtener actualizaciones periódicas de la posición geográfica del dispositivo. Esto se hace mediante la creación de un “escuchador” LocationListener que obtiene la posición mediante el GPS y el registro del mismo con la variable de tipo LocationManager del paso anterior mediante el método requestLocationUpdates, donde LOCATION\_INTERVAL es el intervalo de petición de actualizaciones (un minuto) y LOCATION\_DISTANCE es la distancia de petición de actualizaciones (cien metros):

```
LocationListener locListener=new LocationListener(LocationManager.GPS_PROVIDER);
locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER,LOCATION_INTERVAL,
LOCATION_DISTANCE, locListener);
```

Este LocationListener es una clase que se ha creado implementando la clase android.location.LocationListener y sobrescribiendo algunos de sus métodos, entre ellos onLocationChanged para que al detectar un cambio de posición ésta se inserte en la base de datos de coordenadas.

El método onLocationChanged (public void onLocationChanged(Location location)) se basa en obtener la longitud y la latitud de la posición recibida (location) y compararla con la última calculada (-1, -1 inicialmente). Si la posición no dista más de cien metros de la anterior calculada, se incrementa un contador de repeticiones que simbolizan los minutos que ha estado el técnico en una misma localización. De lo contrario, se realiza la inserción de la posición calculada así como de la posición anterior que no se había insertado aún en la tabla tracking de la base de datos. Esta inserción se realiza mediante el uso del servicio servicioInsertarGeo.php.

### 4.4.3 Diagramas

#### 4.4.3.1 Máquina de estados

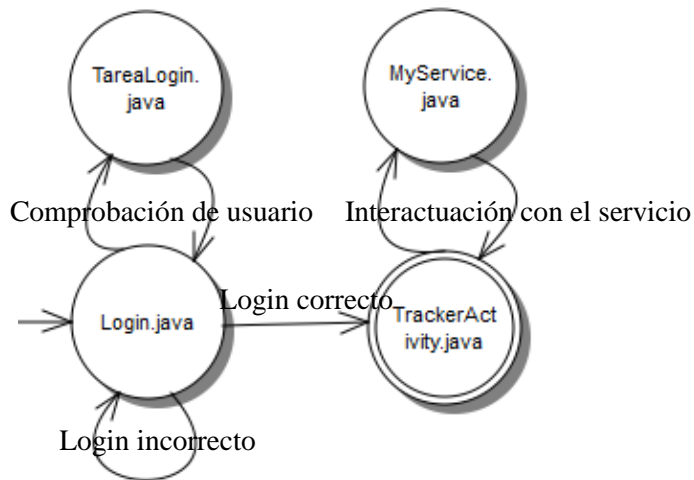


Figura 47. Aplicación Android: Máquina de estados.

#### 4.4.3.2 Diagrama UML

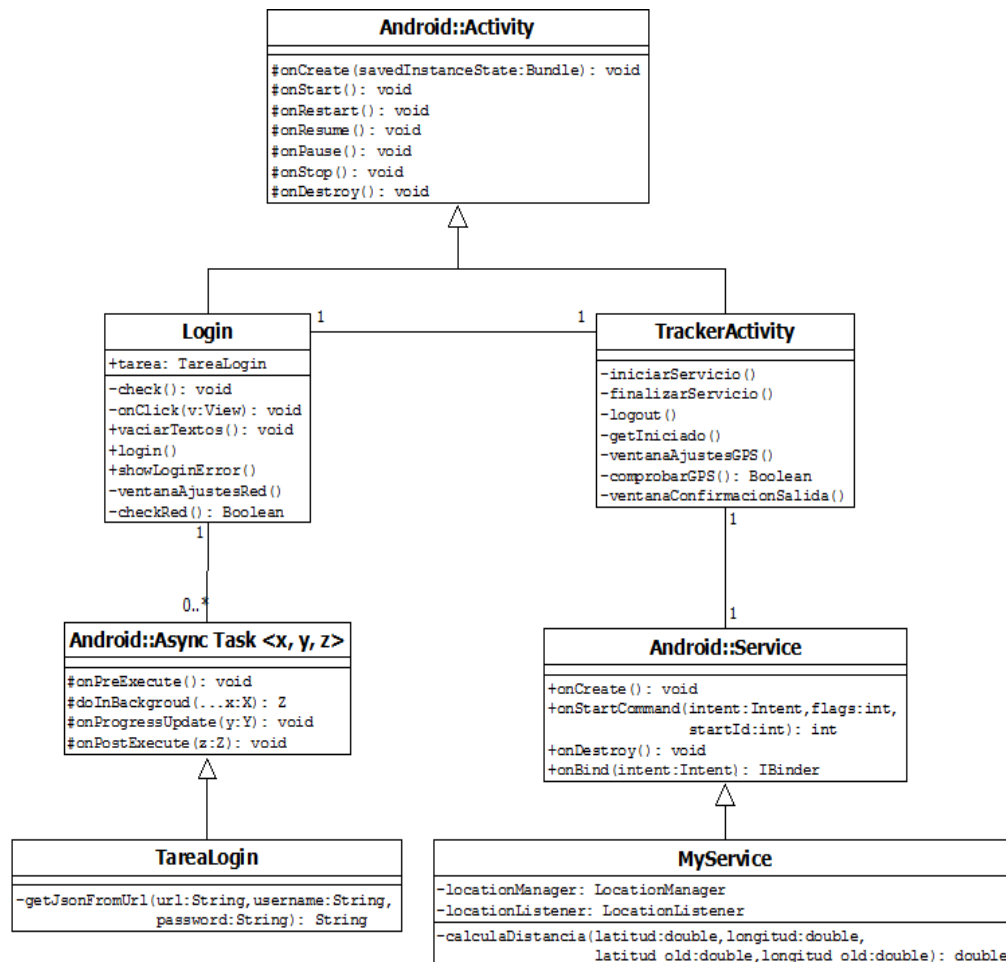


Figura 48. Aplicación Android: Diagrama UML.

#### 4.4.3.3 Diagrama de secuencia

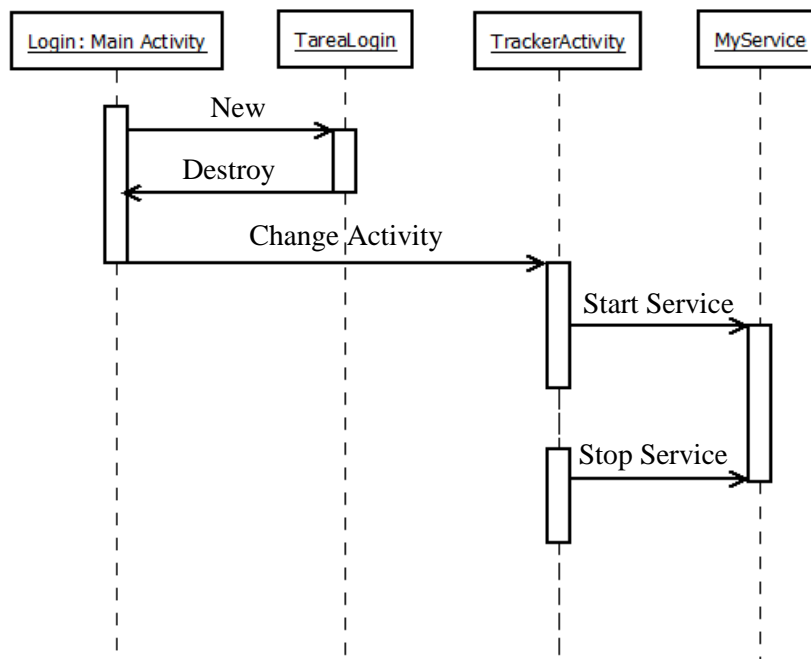


Figura 49. Aplicación Android: Diagrama de secuencia.



## **5 Integración, pruebas y resultados**

---

Para probar el correcto funcionamiento de la aplicación se han realizado pruebas individuales a cada componente del proyecto (aplicación cliente C#, página Web y aplicación móvil de tracking) y pruebas de integración de conjunto.

Para realizar estas pruebas ha sido necesario contratar un hosting para simular el comportamiento de la aplicación a pequeña escala. Este hosting incluye el soporte para poder alojar una base de datos SQLServer de hasta 20 MB y el dominio “albaproyectos.com” donde poder alojar la página Web. Además se ha contratado un buzón de correo que permite dar de alta hasta 400 usuarios.

### **5.1 Pruebas modulares**

#### **5.1.1 Pruebas de aplicación cliente de gestión**

##### **5.1.1.1 Entorno de prueba**

Para realizar esta prueba se necesita un entorno con las siguientes características:

- Equipos con sistemas operativo Windows, preferiblemente la versión 7.
- SQLServer 2012 y Management Studio
- .NET Framework 4 o superior.
- Visual Studio 2010
- Infragistics versión 2011.2
- Navegador de Internet con soporte para Javascript.

##### **5.1.1.2 Pruebas realizadas**

###### **5.1.1.2.1 Pruebas de correcto almacenamiento en la base de datos**

###### **5.1.1.2.1.1. Diseño de la prueba**

La prueba consiste en la consulta de la base de datos mediante sentencias SQL utilizando la herramienta SQL Server Management Studio para comprobar el alta, baja o modificación de registros de la base de datos.

Un ejemplo de prueba sería la utilización del formulario de gestión de empleados por parte del administrador y la comprobación de la correcta realización de las acciones de alta, baja y modificación, comprobando su repercusión en la base de datos.

### 5.1.1.2.1.2. Análisis de los resultados

Al iniciar la prueba el formulario y la tabla de empleados se encuentran de la siguiente forma:

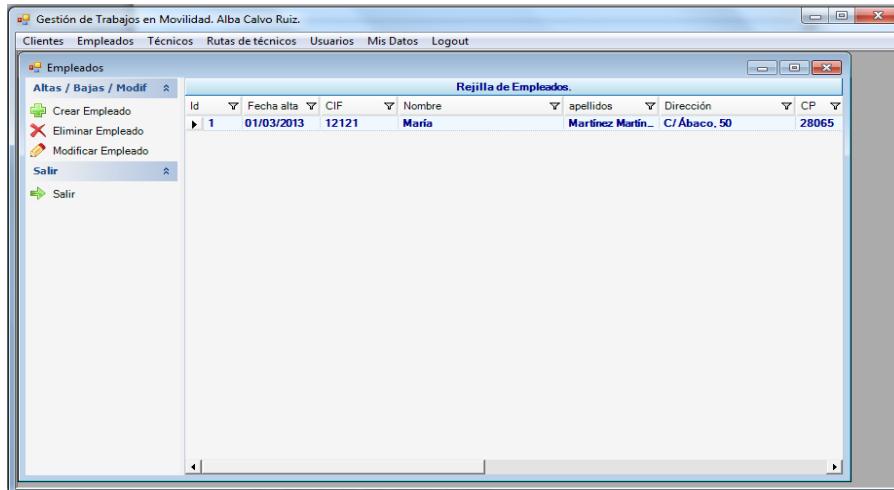


Figura 50. Pruebas modulares. Prueba de aplicación cliente: Rejilla de empleados.

```
select * from empleado;
```

	id	fecha_alta	cf	nombre	apellidos	direccion	d_postal	localidad	provincia	tif	tif_movil	email	coment...	id_usuario
1	1	2013-02-28...	99999	Alba	Calvo Ruiz	C/ Pez, 50	28760	Tres Cantos	Madrid	91999999999	999999999	alba.calvo@albaproyectos.com		1
2	2	2013-03-01...	12121	María	Martínez Martínez	C/ Ábaco, 50	28065	Coslada	Madrid	91121121212	633121121	e.maria.martinez1@albaproyectos.com		4

Figura 51. Pruebas modulares. Prueba de aplicación cliente: Consulta de empleados.

Como puede apreciarse en el cuadro superior, en la tabla empleados se encuentra el administrador, que no se muestra en la ventana de empleados de la aplicación, y una empleada creada previamente.

```
select * from usuario;
```

	id	nombre	apellidos	nick	password	is_admin	is_tecnico
1	1	Alba	Calvo Ruiz	admin	admin	1	0
2	2	Pepe	Pérez Pérez	pepe.perez1	1234	0	1
3	3	Juan	Jiménez Jiménez	juan.jimenez2	1234	0	1
4	4	María	Martínez Martínez	e.maria.martinez1	1234	0	0

Figura 52. Pruebas modulares. Prueba de aplicación cliente: Consulta de usuarios.

La tabla empleados está estrechamente relacionada con la tabla usuario, pues la creación de un empleado supone la creación de su usuario asociado. Esta tabla se encuentra inicialmente como en la imagen anterior.

A continuación se analizará el resultado de las acciones de la barra lateral izquierda del formulario:

- Crear empleado:

The screenshot shows a web form titled 'Ficha de técnico'. It contains the following fields and values:

- Id:** (empty)
- CIF:** 70547645T
- Fecha Alta:** 19/05/13
- Nombre:** Diego
- Apellidos:** Díaz Domínguez
- Dirección:** s/Descubridores, 23
- Localidad:** Tres Cantos
- Provincia:** Madrid
- CP:** 28760
- Teléfono:** 918036586
- Móvil:** 636543472
- Email:** (empty)
- Comentarios:** Alta de nuevo trabajador de oficina

At the bottom, there are two buttons: 'Grabar' and 'Salir'.

**Figura 53. Pruebas modulares. Prueba de aplicación cliente: Alta de empleado.**

```
select * from empleado;
```

	id	fecha_alta	cf	nom...	apellidos	direccion	d_po...	localidad	provincia	tif	tif_movil	email	comentarios	id_us...
1	1	2013-02-28...	99999	Alba	Calvo Ruiz	C/ Pez, 50	28760	Tres Cantos	Madrid	9199999...	999999999	alba.calvo@albaproyectos.com		1
2	2	2013-03-01...	12121	María	Martínez Martí...	C/ Ábaco, 50	28065	Coslada	Madrid	9112112...	633121121	e.maria.martinez1@albaproyecto...		4
3	3	2013-05-19...	70547645T	Diego	Díaz Domínguez	s/Descubridores, 23	28760	Tres Cantos	Madrid	918036586	636543472	e.diego.diaz3@albaproyectos.com	Alta de nuevo...	5

**Figura 54. Pruebas modulares. Prueba de aplicación cliente: Consulta de empleados con cambios reflejados.**

```
select * from usuario;
```

	id	nombre	apellidos	nick	password	is_admin	is_tecnico
1	1	Alba	Calvo Ruiz	admin	admin	1	0
2	2	Pepe	Pérez Pérez	pepe.perez1	1234	0	1
3	3	Juan	Jiménez Jiménez	juan.jimenez2	1234	0	1
4	4	María	Martínez Martínez	e.maria.martinez1	1234	0	0
5	5	Diego	Díaz Domínguez	e.diego.diaz3	1234	0	0

**Figura 55. Pruebas modulares. Prueba de aplicación cliente: Consulta de usuarios con cambios reflejados.**

En las imágenes superiores se puede comprobar que los cambios han surtido efecto en la base de datos y que tanto el empleado como el usuario asociado se han insertado con éxito. Además, ambos formularios se actualizan consecuentemente:

Id	Fecha alta	CIF	Nombre	apellidos	Dirección	CP
3	19/05/2013	70547645T	Diego	Díaz Domínguez	s/Descubridores, 23	28760
2	01/03/2013	12121	María	Martínez Martínez	C/ Ábaco, 50	28065

**Figura 56. Pruebas modulares. Prueba de aplicación cliente: Regilla de empleados con cambios reflejados.**

Id	Nombre	Apellidos	Nick	Password	Administrador?	Tecnico?
5	Diego	Díaz Domínguez	e.diego.diaz3	1234	<input type="checkbox"/>	<input type="checkbox"/>
4	María	Martínez Martínez	e.maria.martinez1	1234	<input type="checkbox"/>	<input type="checkbox"/>
3	Juan	Jiménez Jiménez	juan.jimenez2	1234	<input type="checkbox"/>	<input checked="" type="checkbox"/>
2	Pepé	Pérez Pérez	pepe.perez1	1234	<input type="checkbox"/>	<input checked="" type="checkbox"/>

**Figura 57. Pruebas modulares. Prueba de aplicación cliente: Regilla de usuarios con cambios reflejados.**

- Modificar empleado

Form fields and values:

- Id: 3
- CIF: 70547645T
- Fecha Alta: 19/05/13
- Nombre: Diego
- Apellidos: Díaz Domínguez
- Dirección: s/Descubridores, 23
- Localidad: Tres Cantos
- Provincia: Madrid
- CP: 28760
- Teléfono: 918036586
- Móvil: 636543472
- Email: e.diego.diaz3@albaproyectos.com
- Comentarios: Alta de nuevo trabajador de oficina.

**Figura 58. Pruebas modulares. Prueba de aplicación cliente: Modificación de empleado.**

Los cambios se realizan de forma exitosa tanto en la base de datos como en los formularios asociados:

	id	fecha_alta	cif	nombre	apellidos	direccion	d_postal	localidad	provincia	tif	tif_movil	email
1	1	2013-02-28...	99999	Alba	Calvo Ruiz	C/ Pez, 50	28760	Tres Cantos	Madrid	91999999999	9999999999	alba.calvo@albaproyec
2	2	2013-03-01...	12121	María	Martínez Martínez	C/ Ábaco, 50	28065	Coslada	Madrid	91121121212	633121121	e.maria.martinez1@alb
3	3	2013-05-19...	70547645T	Diego	<u>Díaz Domínguez</u>	s/Descubridores, 23	28760	Tres Cantos	Madrid	918036586	636543472	diego.diaz3@albaproye

**Figura 59. Pruebas modulares. Prueba de aplicación cliente: Consulta de empleados con cambios reflejados 2.**

select \* from usuario;

	id	nombre	apellidos	nick	password	is_admin	is_tecnico
1	1	Alba	Calvo Ruiz	admin	admin	1	0
2	2	Pepe	Pérez Pérez	pepe.perez1	1234	0	1
3	3	Juan	Jiménez Jiménez	juan.jimenez2	1234	0	1
4	4	María	Martínez Martínez	e.maria.martinez1	1234	0	0
5	5	Diego	Díaz-Damasco	diego.diaz3	1234	0	0

**Figura 60. Pruebas modulares. Prueba de aplicación cliente: Consulta de usuarios con cambios reflejados 2.**

Empleados

Altas / Bajas / Modif

Crear Empleado  
Eliminar Empleado  
Modificar Empleado  
Salir

Rejilla de Empleados.							
	Id	Fecha alta	CIF	Nombre	apellidos	Dirección	CP
▶	3	19/05/2013	70547645T	Diego	Díaz-Damasco	s/Descubridores, 23	28760
	2	01/03/2013	12121	María	Martínez Martínez	C/ Ábaco, 50	28065

Salir

**Figura 61. Pruebas modulares. Prueba de aplicación cliente: Rejilla de empleados con cambios reflejados 2.**

Usuarios

Modificaciones

Modificar Usuario  
Salir

Rejilla de Usuarios.							
	Id	Nombre	Apellidos	Nick	Password	Administrador?	Tecnico?
▶	5	Diego	Díaz-Damasco	diego.diaz3	1234	<input type="checkbox"/>	<input type="checkbox"/>
	4	María	Martínez Martínez	e.maria.martinez1	1234	<input type="checkbox"/>	<input type="checkbox"/>
	3	Juan	Jiménez Jiménez	juan.jimenez2	1234	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	2	Pepe	Pérez Pérez	pepe.perez1	1234	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	1	Alba	Calvo Ruiz	admin	admin	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Salir

**Figura 62. Pruebas modulares. Prueba de aplicación cliente: Rejilla de usuarios con cambios reflejados 2.**

- Eliminar empleado

Ficha de técnico

Id: 3

CIF: 70547645T

Fecha Alta: 19/05/13

Nombre: Diego

Apellidos: Díaz Damasco

Dirección: s/Descubridores, 23

Localidad: Tres Cantos

Provincia: Madrid

CP: 28760

Teléfono: 918036586

Móvil: 636543472

Email: diego.diaz3@slbaproyectos.com

Comentarios: Alta de nuevo trabajador de oficina.

Eliminar Salir

**Figura 63. Pruebas modulares. Prueba de aplicación cliente: Eliminación de empleado.**

Los cambios se realizan de forma exitosa tanto en la base de datos como en los formularios asociados:

```
select * from empleado;
```

100 %

Resultados Mensajes

	id	fecha_alta	cif	nombre	apellidos	direccion	d_postal	localidad	provincia	tif	tif_movil	email
1	1	2013-02-28...	99999	Alba	Calvo Ruiz	C/ Pez, 50	28760	Tres Cantos	Madrid	91999999999	999999999	alba.calvo@alba
2	2	2013-03-01...	12121	María	Martínez Martínez	C/ Ábaco, 50	28065	Coslada	Madrid	91121121212	633121121	e.maria.martinez1

**Figura 64. Pruebas modulares. Prueba de aplicación cliente: Consulta de empleados con cambios reflejados 3.**

```
select * from usuario;
```

100 %

Resultados Mensajes

	id	nombre	apellidos	nick	password	is_admin	is_tecnico
1	1	Alba	Calvo Ruiz	admin	admin	1	0
2	2	Pepe	Pérez Pérez	pepe.perez1	1234	0	1
3	3	Juan	Jiménez Jiménez	juan.jimenez2	1234	0	1
4	4	María	Martínez Martínez	e.maria.martinez1	1234	0	0

**Figura 65. Pruebas modulares. Prueba de aplicación cliente: Consulta de usuarios con cambios reflejados 3.**

Empleados

Altas / Bajas / Modif

Crear Empleado

Eliminar Empleado

Modificar Empleado

Salir

Salir

Rejilla de Empleados.

Id	Fecha alta	CIF	Nombre	apellidos	Dirección	CP
2	01/03/2013	12121	María	Martínez Martín	C/ Ábaco, 50	28065

**Figura 66. Pruebas modulares. Prueba de aplicación cliente: Rejilla de empleados con cambios reflejados 3.**

Usuarios

Modificaciones

Modificar Usuario

Salir

Salir

Rejilla de Usuarios.

Id	Nombre	Apellidos	Nick	Password	Administrador?	Tecnico?
4	María	Martínez Martínez	e.maria.martine_	1234	<input type="checkbox"/>	<input type="checkbox"/>
3	Juan	Jiménez Jiménez	juan.jimenez2	1234	<input type="checkbox"/>	<input checked="" type="checkbox"/>
2	Pepe	Pérez Pérez	pepe.perez1	1234	<input type="checkbox"/>	<input checked="" type="checkbox"/>
1	Alba	Calvo Ruiz	admin	admin	<input checked="" type="checkbox"/>	<input type="checkbox"/>

**Figura 67. Pruebas modulares. Prueba de aplicación cliente: Rejilla de usuarios con cambios reflejados 3.**

### 5.1.1.2.2 Pruebas de interfaz

#### 5.1.1.2.2.1. Diseño de la prueba

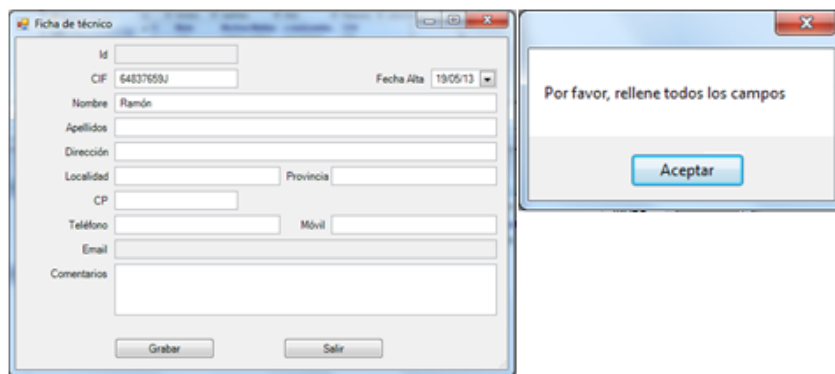
La prueba consiste en la comprobación del correcto funcionamiento de cada uno de los componentes de la interfaz, incluidos los campos de texto con limitación en el número de caracteres.

#### 5.1.1.2.2.2. Análisis de resultados

El correcto funcionamiento de los botones y las rejillas de datos se ha comprobado mediante la ejecución de todos los flujos posibles y el resultado ha sido correcto.

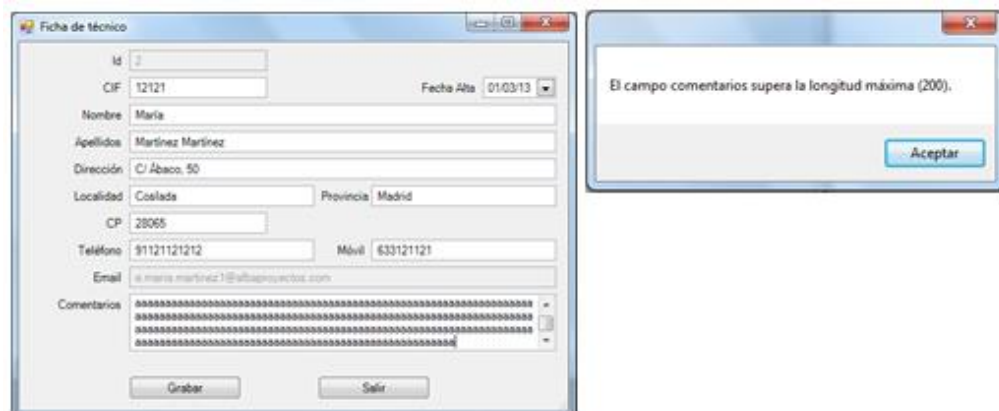
También se ha comprobado en los formularios de alta, baja y modificación la completitud de los campos esenciales y la longitud máxima de los mismos, mostrando un mensaje al usuario en caso de alguna anomalía.

- Campos sin rellenar:



**Figura 68. Pruebas modulares. Prueba de aplicación cliente: Prueba de campos sin rellenar.**

- Campos demasiado largos:



**Figura 69. Pruebas modulares. Prueba de aplicación cliente: Prueba de campos demasiado largos.**

## 5.1.2 Pruebas de aplicación Web

### 5.1.2.1 Entorno de prueba

Para realizar esta prueba se necesita un entorno con las siguientes características:

- PHP5
- Librerías JQuery
- Navegador de Internet con soporte para Javascript.

### 5.1.2.2 Pruebas realizadas

#### 5.1.2.2.1 Pruebas de flujo

##### 5.1.2.2.1.1. Diseño de la prueba

La prueba consiste en la ejecución de todos los posibles flujos correspondientes a las acciones de los usuarios de la aplicación Web.

##### 5.1.2.2.1.2. Análisis de resultados

Se ha comprobado que el funcionamiento de la aplicación Web es correcto y que soporta toda la funcionalidad deseada.

#### 5.1.2.2.2 Pruebas de notificación de errores

##### 5.1.2.2.2.1. Diseño de la prueba

La prueba consiste en la ejecución de todas las posibles acciones que puedan generar un error (login incorrecto, inexistencia de avisos sin leer...) y comprobar su correcta notificación al usuario.

##### 5.1.2.2.2.2. Análisis de resultados

Se ha comprobado que los posibles errores se notifican correctamente al usuario:

- Campos sin rellenar en el formulario de login



**Figura 70. Pruebas modulares. Prueba de aplicación Web: Prueba de campos sin rellenar en el formulario de login.**

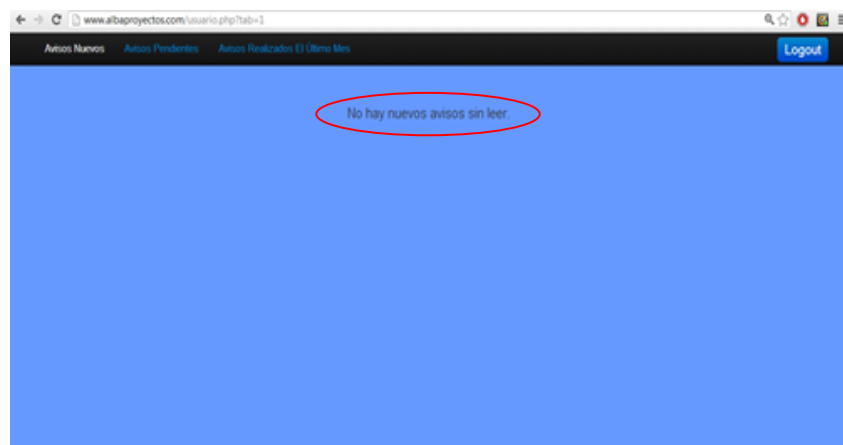


- Login incorrecto



**Figura 71. Pruebas modulares. Prueba de aplicación Web: Prueba login incorrecto.**

- Ausencia de avisos sin leer en la base de datos:



**Figura 72. Pruebas modulares. Prueba de aplicación Web: Prueba de ausencia de avisos sin leer en la base de datos.**

### 5.1.3 Pruebas de aplicación de geolocalización

#### 5.1.3.1 Entorno de prueba

Para realizar esta prueba se necesita un entorno con las siguientes características:

- Smartphone con sistema operativo Android (pruebas realizadas con versión Gingerbread 2.3.3 y superiores).

### **5.1.3.2 Pruebas realizadas**

#### **5.1.3.2.1 Pruebas de acceso a servicios Web**

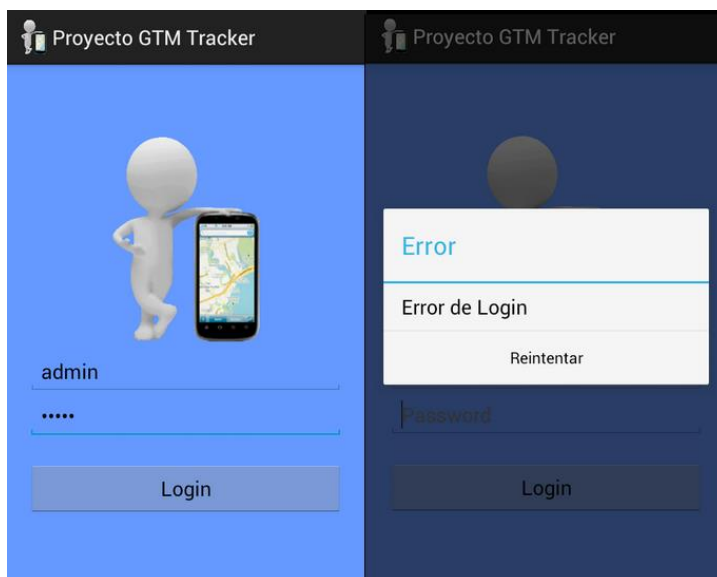
##### **5.1.3.2.1.1. Diseño de la prueba**

La prueba consiste en la comprobación del correcto acceso al servicio Web de login alojado en el servidor de la página Web.

##### **5.1.3.2.1.2. Análisis de resultados**

Se ha comprobado que la tarea asíncrona encargada de realizar el login consulta correctamente la tabla de usuarios de la base de datos y permite el acceso sólo a los usuarios con rol de técnico mediante la utilización de un servicio Web que responde en forma de fichero JSON.

El resultado ha sido el esperado: en caso de nick o contraseña incorrectos o de usuario no autorizado, se muestra un mensaje de informe de error:



**Figura 73. Pruebas modulares. Prueba de aplicación de geolocalización: Prueba de login.**

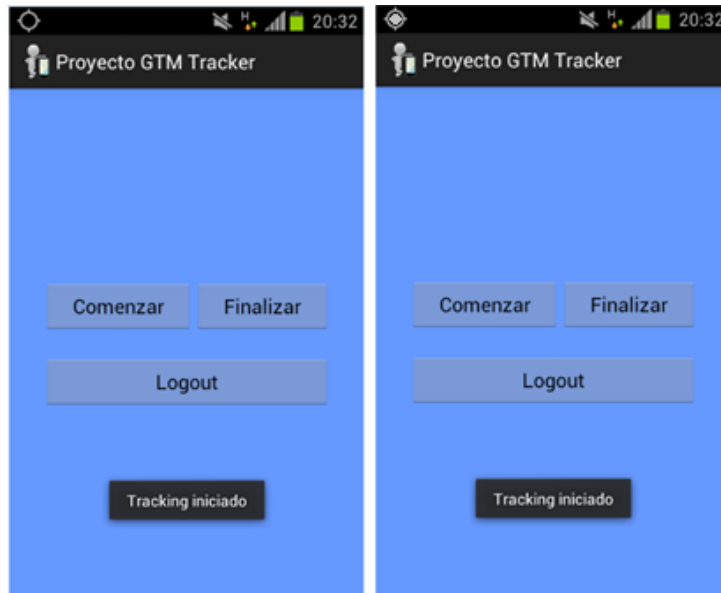
#### **5.1.3.2.2 Pruebas de servicio de geolocalización**

##### **5.1.3.2.2.1. Diseño de la prueba**

La prueba consiste en la comprobación de la correcta ejecución del servicio, de su permanencia aunque la aplicación principal se cierre y del correcto almacenamiento de las coordenadas registradas en la base de datos.

#### 5.1.3.2.2.2. *Análisis de resultados*

Se ha comprobado la correcta iniciación y detención del servicio al pulsar los botones correspondientes mediante la aparición de un icono parpadeante que simboliza el tracking en la barra superior de la aplicación:



**Figura 74. Pruebas modulares. Prueba de aplicación de geolocalización: Prueba de servicio de geolocalización.**

Además, se ha comprobado la persistencia del servicio al pasar la aplicación a segundo plano o ser cerrada por el planificador de procesos.

Por último, se ha comprobado el correcto almacenamiento en la bases de datos de las coordenadas mediante la realización de múltiples rutas poniéndose en el lugar de un técnico. El resultado de una de ellas en base de datos se muestra en la imagen inferior, donde las repeticiones mayores de 5 representan las paradas del técnico:

	id	tecnico_id	longitud	latitud	fecha	repeticiones
1	1	1	-3.701856	40.59934	2013-04-17 09:15:05.0000000	0
2	2	1	-3.70132	40.596309	2013-04-17 09:16:15.0000000	7
3	3	1	-3.70677	40.595495	2013-04-17 09:24:15.0000000	0
4	4	1	-3.707607	40.599209	2013-04-17 09:25:15.0000000	0
5	5	1	-3.705053	40.600725	2013-04-17 09:26:15.0000000	8
6	6	2	-3.705053	40.620725	2013-04-17 09:26:15.0000000	0

**Figura 75. Pruebas modulares. Prueba de aplicación de geolocalización: Consulta de coordenadas de geolocalización almacenadas en la base de datos.**

## **5.2 Pruebas de integración**

### **5.2.1 Diseño de la prueba**

Tras probar cada uno de los módulos por separado se ha procedido a comprobar que la interacción entre ellos era correcta. Por ejemplo: los avisos asociados a un técnico dados de alta en la aplicación en C# deben aparecer en la página Web cuando éste se identifica, las coordenadas registradas en la aplicación Android deben poder ser trasladadas a un mapa de Google Maps y presentadas al usuario en la Web, etc.

### **5.2.2 Análisis de resultados**

Todos los módulos probados previamente acceden a una base de datos común y realizan cambios en ella.

Dado que todos ellos funcionaban correctamente por separado reflejando sus cambios en la misma, no ha habido ningún problema de integración.

## 6 Conclusiones y trabajo futuro

---

### 6.1 Conclusiones

La gestión de empresas con actividad de SAT y, más concretamente, la implantación de servicios de movilidad dentro de las mismas, se ha identificado como una necesidad real que el proyecto podría abordar.

En respuesta a esa necesidad se ha desarrollado una solución de gestión de avisos en movilidad con los siguientes componentes:

- Aplicación cliente Windows de gestión de avisos.
- Aplicación Web para la gestión de avisos en movilidad y la representación cartográfica de ubicaciones y rutas de técnicos.
- Aplicación cliente Android para el tracking GPS de terminales.

La solución, tal y como se ha implementado, es utilizable como producto autónomo por empresas SAT, siendo sus prestaciones más destacadas:

- El envío inmediato de avisos a los técnicos y la grabación por parte de éstos del trabajo realizado en sus terminales.
- La representación cartográfica de ubicaciones y rutas de técnicos en el navegador.

El producto soporta una funcionalidad genérica y cuenta con una arquitectura de fácil implantación. Esto debería permitir la realización de ofertas a un elevado número de empresas. Las ofertas podrían incluir la posibilidad de ampliación para cubrir funcionalidad adicional demandada por el cliente. En función de la realimentación recibida, el producto podría ir evolucionando.

En cuanto a la parte formativa, la elaboración del proyecto me ha permitido tomar contacto con las siguientes tecnologías:

- SQLServer y Management Studio 2012 para la gestión de bases de datos.
- Desarrollo de aplicaciones Windows Forms en C# utilizando controles de Infragistics en la capa de presentación y ORM Microsoft Entity Framework en la capa de acceso a datos.
- Desarrollo de aplicaciones Web utilizando PHP y Javascript con uso de librerías JQuery y de la API de Google Maps.
- Desarrollo de servicios Web para la recepción y escritura en base de datos de coordenadas geográficas y para la autenticación de usuarios.
- Desarrollo de aplicaciones multihilo en Java para Android.
- Creación de paquetes de instalación.

Técnicamente, el mayor reto ha sido la búsqueda de una alternativa orientada a objetos para el mantenimiento de la base de datos frente a SQL. La solución adoptada ha sido el ORM Microsoft Entity Framework.

Otro reto técnico ha sido conseguir continuidad en el servicio de tracking ante posibles cierres de la aplicación principal por el manejador de procesos de Android. El servicio de geolocalización y subida de coordenadas a la base de datos se ha implementado mediante un hilo independiente inmune a cierres de la aplicación principal ajenos al usuario y reutilizable por las nuevas instancias de la aplicación que pudieran lanzarse.

Como conclusiones finales cabe destacar, por un lado, que el proyecto GTM me ha permitido una toma de contacto con el modelo de gestión de empresas SAT y con un variado conjunto de tecnologías necesarias para la implantación, y por otro, que el producto resultante es una solución conectada con el mundo real que podría constituir el embrión de un producto con un fuerte potencial de comercialización en el sector de empresas SAT.

## **6.2 Trabajo futuro**

El proyecto podría ser objeto de mejora en relación con algunas limitaciones en su implementación actual:

- El proyecto da soporte a la gestión de avisos en movilidad de una empresa SAT. El alcance del proyecto no incluye aspectos de la actividad SAT como mantenimiento preventivo de instalaciones, revisiones periódicas, etc. Además, la funcionalidad del aviso podría ser objeto de mejora.
- La aplicación cliente de gestión presenta las limitaciones típicas de este tipo de solución en relación con la posibilidad de trabajar de manera deslocalizada y con la necesidad de realizar mantenimiento en el PC de cada usuario. No obstante, en pequeñas empresas estas limitaciones se atenúan y pierden peso frente a la ventaja de una mayor productividad en el desarrollo de la interfaz de usuario en el entorno de aplicación cliente.
- La aplicación Web de visualización y cumplimentación de avisos por técnicos no permite el trabajo offline en situaciones de pérdida de conectividad ni el futuro soporte de funcionalidad avanzada como escaneo de la firma del cliente, etc.
- La aplicación Web en PHP presenta las limitaciones de escalabilidad frente a Java en un hipotético escenario de participación de múltiples desarrolladores en el proyecto.
- Se utiliza Google Maps con licencia gratuita que permite un máximo de 2.500 solicitudes diarias. Dependiendo del volumen de cada empresa, podría ser necesario adquirir la licencia “Google Maps for Business”. Por otra parte, los mapas Web disponibles en la actual implantación no pueden integrarse con la aplicación cliente de gestión Windows, quedando descartadas prestaciones como la visualización de localización de los técnicos dentro del formulario de recepción de un aviso, etc.

En relación con las limitaciones anteriores y con el fin de conseguir un producto más competitivo, se podrían introducir mejoras en el proyecto en aspectos como:

- Ampliación de funcionalidad de la aplicación Web de recepción y cumplimentación de avisos por técnicos con aspectos adicionales como subida de fotografías, etc.
- Desarrollo de un nuevo módulo de gestión documental donde podrían integrarse las fotografías recibidas de los técnicos junto con otros ficheros como firmas escaneadas de clientes, planos de instalaciones, etc.
- Ampliación de la funcionalidad de representación cartográfica con incorporación de nuevos mapas como localización de avisos pendientes de realizar, etc.
- Desarrollo de aplicación cliente Android como complemento a la actual aplicación Web de recepción y cumplimentación de avisos por parte de los técnicos. La aplicación cliente podría ofrecer prestaciones como:
  - Trabajo offline en situaciones de pérdida de conectividad.
  - Notificación a los técnicos de la llegada de nuevos avisos mediante un mecanismo más efectivo que la actual notificación vía email.
  - Escaneo de firmas de clientes a la finalización del aviso y subida de las mismas para su integración en la aplicación de gestión.<sup>1</sup>
  - Impresión de boletines/facturas utilizando impresoras Bluetooth o integradas en el terminal.
- Ampliación de funcionalidad para dar un mayor soporte a la gestión de una empresa SAT, dando respuesta a necesidades como registro de instalaciones objeto de mantenimiento preventivo o correctivo, planificación automática de tareas periódicas, reparto automático de avisos recibidos, estadísticas, etc.
- Desarrollo de un módulo financiero (facturación, contabilidad, etc.) o integración con productos de terceros. Las facturas emitidas y otros documentos con o sin firma digital podrían integrarse en el módulo de gestión documental citado anteriormente.
- Extensión de la aplicación cliente para terminales en movilidad a otros sistemas operativos diferentes a Android.
- Integración del módulo de movilidad con aplicaciones sectoriales de terceros que no cuenten con servicio de movilidad, previo acuerdo con sus desarrolladores.

Aunque de algunas de las ampliaciones propuestas se derivan incertidumbres técnicas, existe una gran relación entre éstas y la implementación actual del proyecto GTM, de la cual son una evolución.

Gracias al bagaje adquirido durante la elaboración del proyecto, el desarrollo de las mejoras anteriores sería factible invirtiendo, claro está, el tiempo y esfuerzo necesarios.





# Bibliografía

---

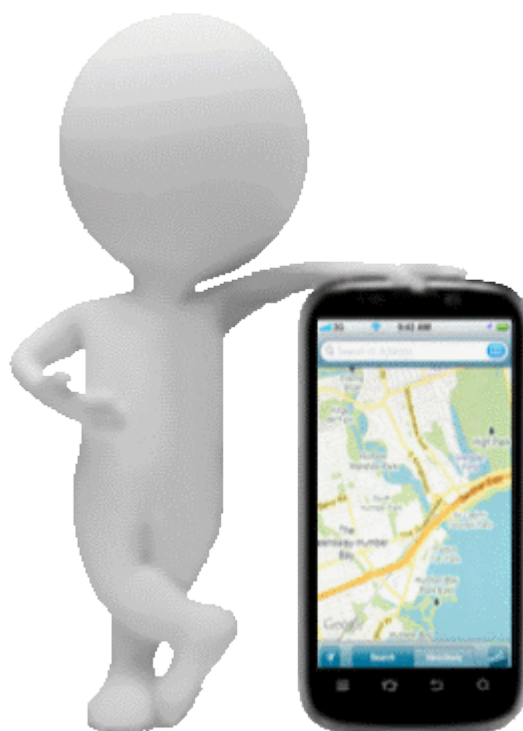
- [1] Natalia Arroyo Vázquez, “Informe APEI sobre movilidad”, Gijón: Asociación Profesional de Especialistas en Información, 2011
- [2] Ministerio de Industria, Energía y Turismo, “Tecnologías de movilidad: Análisis sectorial de implantación de las TIC en la PYME española”  
<http://www.fundetec.es/wp-content/uploads/2012/03/Especial-Movilidad-epyme11.pdf>
- [3] Jesús Manuel Plaza Llorente, “La innovación empresarial en perspectiva. Un análisis de datos de panel”, 2012, páginas 119-128
- [4] Raúl Roses, Jaime Patricio Zarate Piray, “Estudio comparativo de las tecnologías Punto Net y Java para el desarrollo de aplicaciones Web”
- [5] María Helena Trejos Arroyave, Diego Fernando Zamora Cardona, “Criterios de evaluación de plataformas de desarrollo de aplicaciones empresariales para entornos Web”, 2012
- [6] Gabriel Herraiz Antón, “ANDROID”, Noviembre 2012, páginas 15-26
- [7] Unnikrishnan.P.K, “Blackberry technology, a seminar report”, School of Engineering COCHIN University Of Science & Technology, KOCHI-682022, August 2008
- [8] Santiago Gómez Ruiz, “Microsoft SQL Server, MySQL y PostgreSQL”
- [9] Daniel Bartholome, “Comparamos MariaDB y MySQL cara a cara”, Linux Magazine, número 73, páginas 44-46.
- [10] Juan Ferrer Martínez, “Implantación de Aplicaciones Web”, 2012, Capítulo 1
- [11] José Barato, Israel Marcos, Javier Cantalapiedra, Carlos Von Prabucki, Prudencio Poza, Ignacio Pérez, “Plataformas de Arquitectura de Aplicaciones. El fin de la discusión J2EE VS .NET”, Atos Origin, Madrid, 2006



## **Anexos**

---

### ***A Manual de instalación***



**ALBAPROYECTOS S.L.**

**MANUAL DE INSTALACIÓN**

**Gestión de Trabajos en Movilidad (GTM)**



## ÍNDICE

1 Introducción.....	iii
2 Requisitos software .....	iii
3 Requisitos de acceso a recursos del servidor.....	iii
4 Posibles escenarios de instalación .....	iii
4.1 Casuística de escenarios .....	iii
4.2 Paquetes a instalar en cada caso. ....	iv
5 Descripción de la instalación de paquetes .....	iv
5.1 Instalación en PCs .....	iv
5.2 Instalación en dispositivos móviles con Android .....	v



# **1 Introducción**

---

En los apartados siguientes se describen los pasos necesarios para la instalación de la aplicación Gestión de Trabajos en Movilidad.

Para realizar la instalación debe disponer de los siguientes paquetes de instalación:

- InstalacionProyectoGTM.exe
- tracker\_servicio.apk

## **2 Requisitos software**

---

Para que la aplicación funcione correctamente es necesario que cada terminal PC de trabajo cumpla los siguientes requisitos:

- Sistema operativo Windows 7.
- .NET Framework 4.0 o superior
- Navegador Web con Javascript habilitado

Asimismo, es necesario que los terminales móviles dispongan de la versión de Android Gingerbread 2.3.3 o superior.

## **3 Requisitos de acceso a recursos del servidor**

---

La aplicación accede a una base de datos en la nube accesible a través de un servidor de base de datos SQLServer 2012.

## **4 Posibles escenarios de instalación**

---

### **4.1 Casuística de escenarios**

Son posibles 3 escenarios de instalación:

- 1) PCs pertenecientes a un usuario con rol de “administrador”.
- 2) PCs pertenecientes a un usuario con rol de “empleado”.
- 3) Terminales móviles pertenecientes a un usuario con rol de “técnico”.

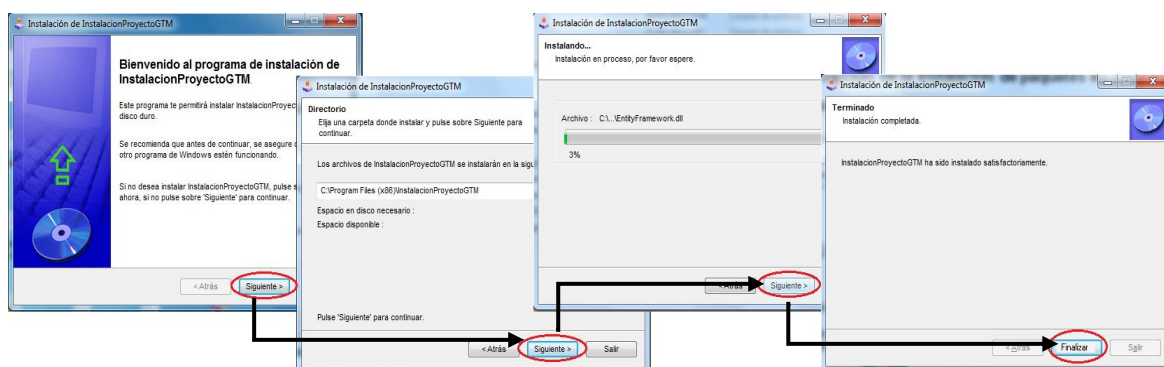
## 4.2 Paquetes a instalar en cada caso.

Caso n°	Tipo de Usuario	Paquetes a instalar
1	Administrador	InstalacionProyectoGTM.exe
2	Empleado	InstalacionProyectoGTM.exe
3	Técnico	tracker_servicio.apk

## 5 Descripción de la instalación de paquetes

### 5.1 Instalación en PCs

Ejecute el paquete de instalación “InstalacionProyectoGTM.exe”:



**Figura 76. Anexo I: Manual de instalación. Paquete de instalación de la aplicación cliente de gestión.**

Una vez finalizada la instalación, aparecerán dos accesos directos a la aplicación en el escritorio y en el menú de inicio:



**Figura 77. Anexo I: Manual de instalación. Accesos directos de la aplicación cliente de gestión.**



## 5.2 Instalación en dispositivos móviles con Android

Copie el fichero “tracker\_servicio.apk” a la memoria interna del teléfono y selecciónelo para llevar a cabo el proceso de instalación:

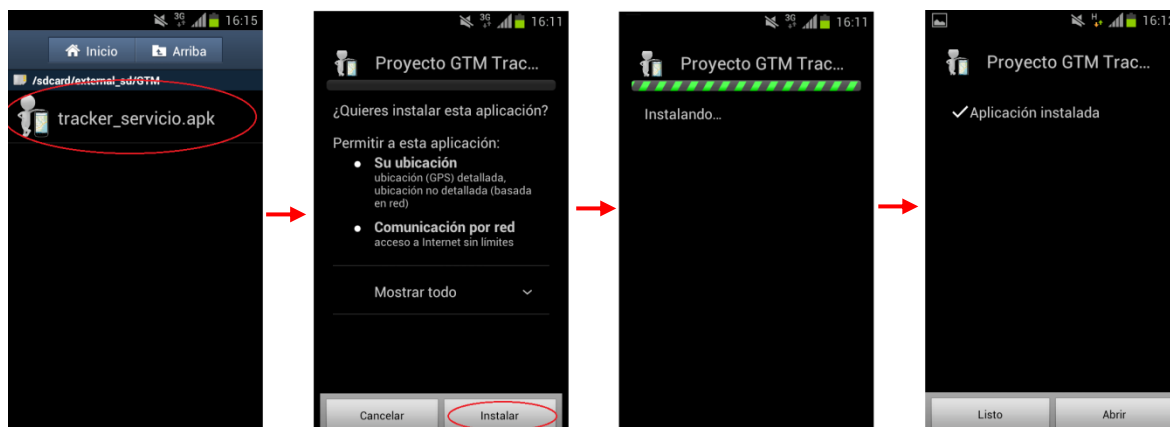
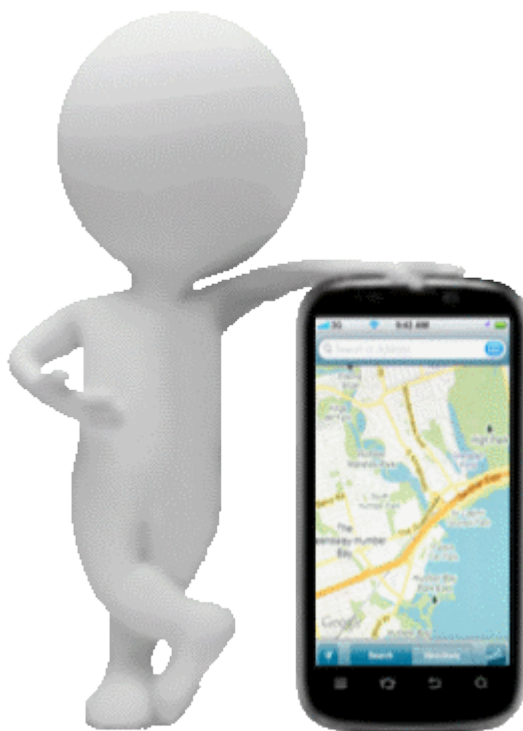


Figura 78. Anexo I: Manual de instalación. Instalación de la aplicación Android.



## ***B Manual de usuario***



**ALBAPROYECTOS S.L.**

**MANUAL DE USUARIO**

**Gestión de Trabajos en Movilidad (GTM)**



## ÍNDICE

1 Introducción.....	iii
2 Estructura del proyecto GTM .....	iii
3 Manual de la aplicación de gestión.....	iv
3.1 Pantalla de identificación.....	iv
3.2 Opciones de administrador .....	iv
3.2.1 Clientes .....	v
3.2.2 Empleados.....	vi
3.2.3 Técnicos .....	vii
3.2.4 Tracking de técnicos .....	vii
3.2.5 Usuarios .....	x
3.2.6 Mis datos.....	x
3.2.7 Logout.....	xi
3.3 Opciones de empleado.....	xi
3.3.1 Clientes .....	xi
3.3.2 Tracking de técnicos .....	xi
3.3.3 Mis datos.....	xi
4 Manual de la aplicación Web .....	xii
4.1 Pantalla de identificación.....	xii
4.2 Opciones de administrador .....	xiii
4.3 Opciones de técnico.....	xiv
4.3.1 Avisos nuevos .....	xv
4.3.2 Avisos pendientes .....	xv
4.3.3 Avisos realizados el último mes .....	xvi
5 Manual de la aplicación de geolocalización .....	xvii
5.1 Pantalla de identificación.....	xvii
5.2 Pantalla de geolocalización .....	xvii



# 1 Introducción

El proyecto GTM es una aplicación informática dirigida a empresas que realizan trabajos de mantenimiento y reparación en domicilios de clientes. La aplicación proporcionaría a las empresas usuarias las siguientes ventajas:

- Eliminaría las tareas de emisión y reparto de partes de trabajo en papel.
- Permitiría conocer en tiempo real la realización de los trabajos, con la consiguiente ventaja de cara a la asignación de nuevos trabajos a técnicos, etc.
- Facilitaría a los operadores de la aplicación cliente información actualizada útil en su interacción con los clientes, permitiendo respuestas como "el técnico ya acudió hace una hora pero no había nadie en casa", etc.
- Evitaría la grabación de datos por el personal de la oficina con el consiguiente ahorro de tiempo y eliminación de errores derivados de la interpretación de los textos manuscritos de los técnicos.

Este documento le permitirá aprender a utilizar todas las funcionalidades básicas de las aplicaciones que componen el proyecto GTM.

## 2 Estructura del proyecto GTM

El diseño del proyecto está compuesto de tres partes fundamentales:

- La aplicación cliente en C#
- La página Web PHP
- La aplicación de localización vía GPS para plataformas Android

Todas ellas acceden a una base de datos común, como se muestra en el siguiente esquema:

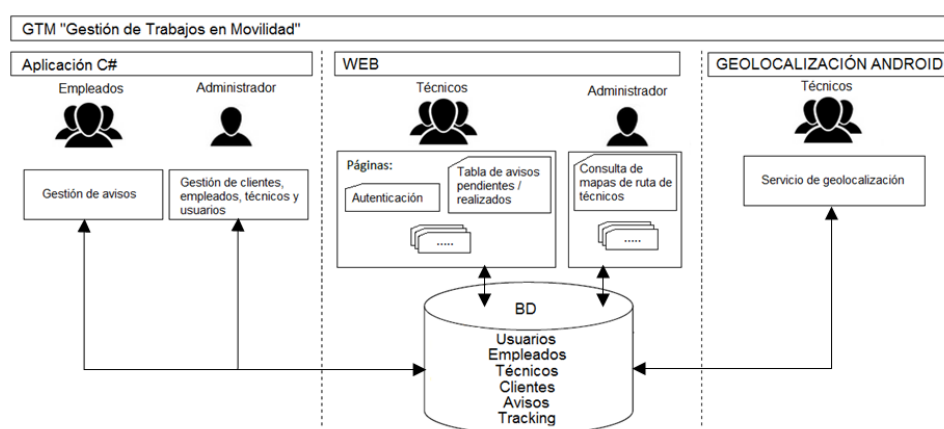


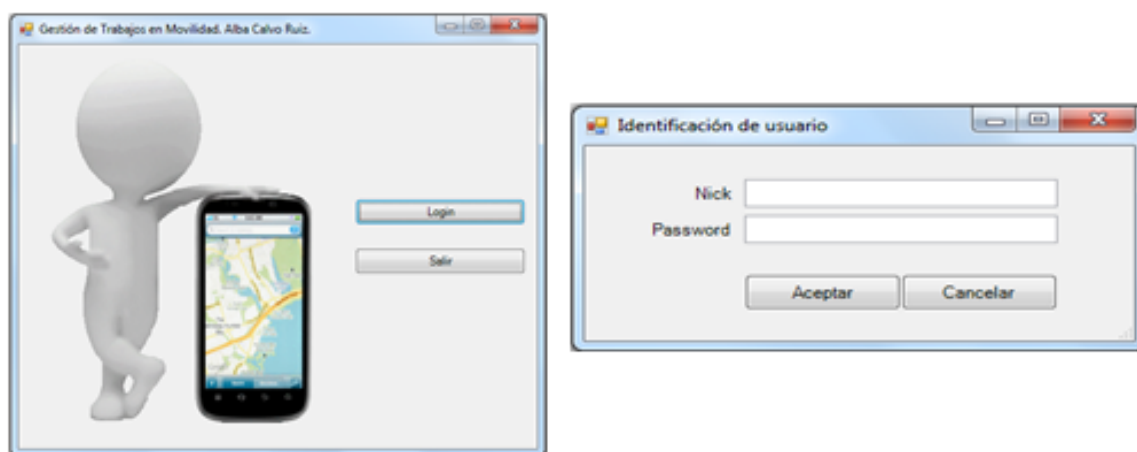
Figura 79. Anexo II: Manual de usuario. Esquema del proyecto GTM.

A continuación se presentarán tres manuales de introducción al uso de cada una de las partes del proyecto.

## 3 Manual de la aplicación de gestión

---

### 3.1 Pantalla de identificación

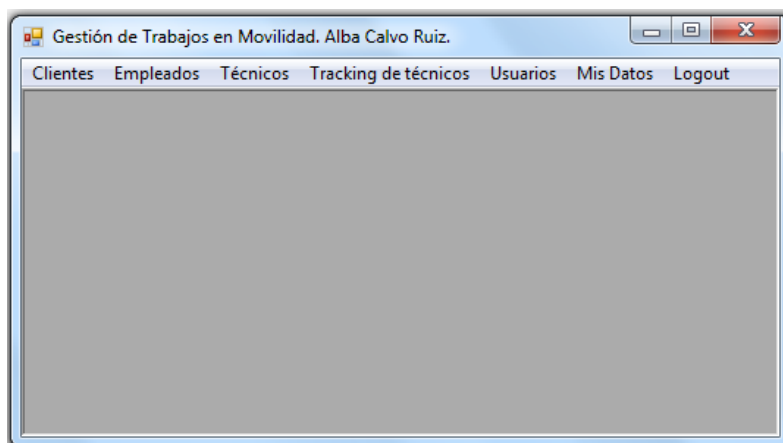


**Figura 80. Anexo II: Manual de usuario. Aplicación de gestión: Ventana de login.**

Lo primero que debe hacer para comenzar a utilizar la aplicación es pulsar el botón “Login” y autenticarse con sus datos de usuario.

La aplicación distingue dos tipos de usuario, administrador y empleado. Si usted tiene el primer rol, diríjase al siguiente apartado (3.2. Opciones de administrador). De lo contrario, diríjase al apartado 3.3 (Opciones de empleado).

### 3.2 Opciones de administrador



**Figura 81. Anexo II: Manual de usuario. Aplicación de gestión: Menú de administrador.**



Tras identificarse, se le mostrará un menú con varias pestañas que representan sus opciones como usuario administrador.

### 3.2.1 Clientes

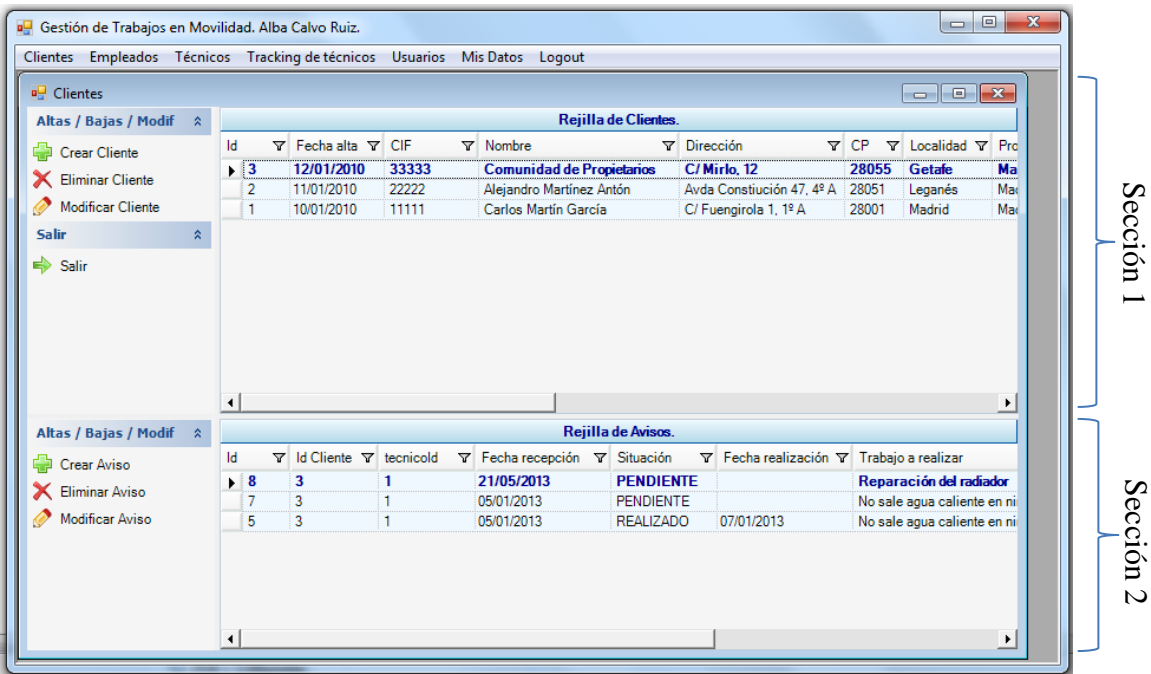


Figura 82. Anexo II: Manual de usuario. Aplicación de gestión: Ventana de Clientes.

La selección de la opción “Clientes” del menú principal abre el formulario correspondiente. El formulario se divide en dos secciones. La primera de ellas presenta un listado de clientes, que podemos ordenar y buscar por varios criterios.

Desde la relación de clientes se pueden realizar altas, bajas y modificaciones de clientes mediante las opciones presentes en la barra lateral izquierda.

Bajo el listado de clientes, en la parte inferior del formulario, se encuentra una rejilla con los avisos del cliente actualmente seleccionado que permite realizar altas, bajas y modificaciones de avisos asociados al cliente seleccionado. Esta rejilla se refresca automáticamente cada vez que cambiamos de cliente.

### 3.2.2 Empleados

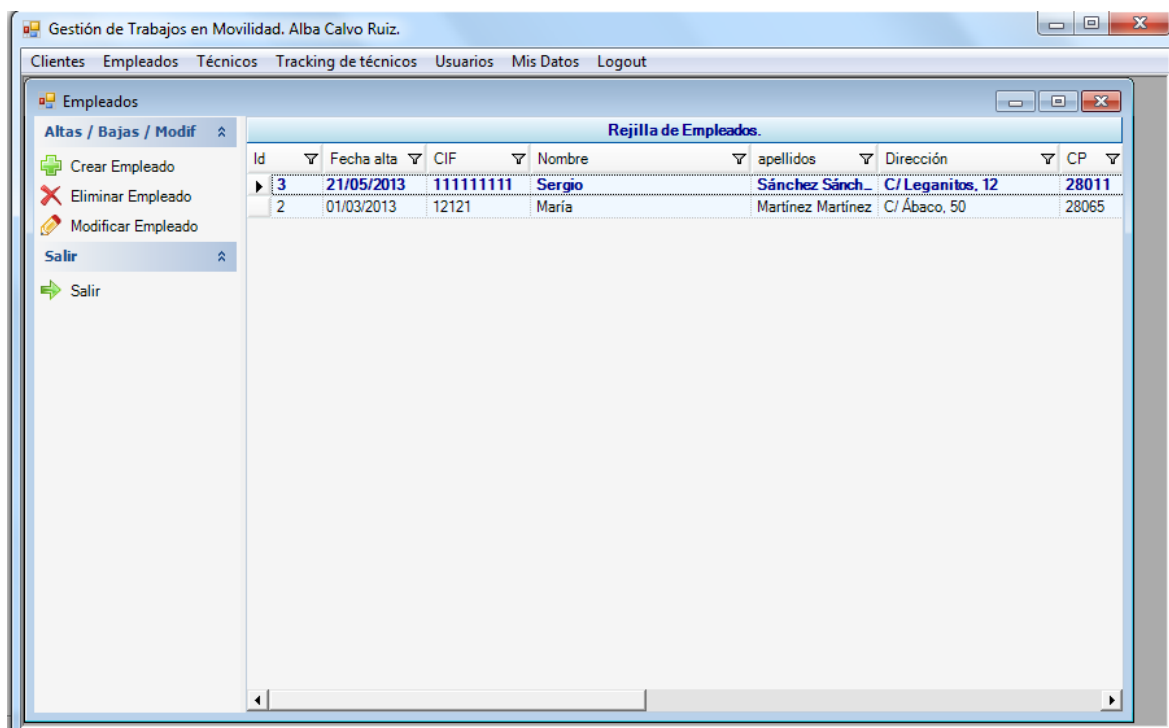


Figura 83. Anexo II: Manual de usuario. Aplicación de gestión: Ventana de Empleados.

La selección de la opción “Empleados” del menú principal abre el formulario correspondiente. El formulario presenta un listado de empleados que podemos ordenar y buscar por varios criterios y además permite la realización de altas, bajas y modificaciones mediante las opciones presentes en la barra lateral izquierda.

### 3.2.3 Técnicos

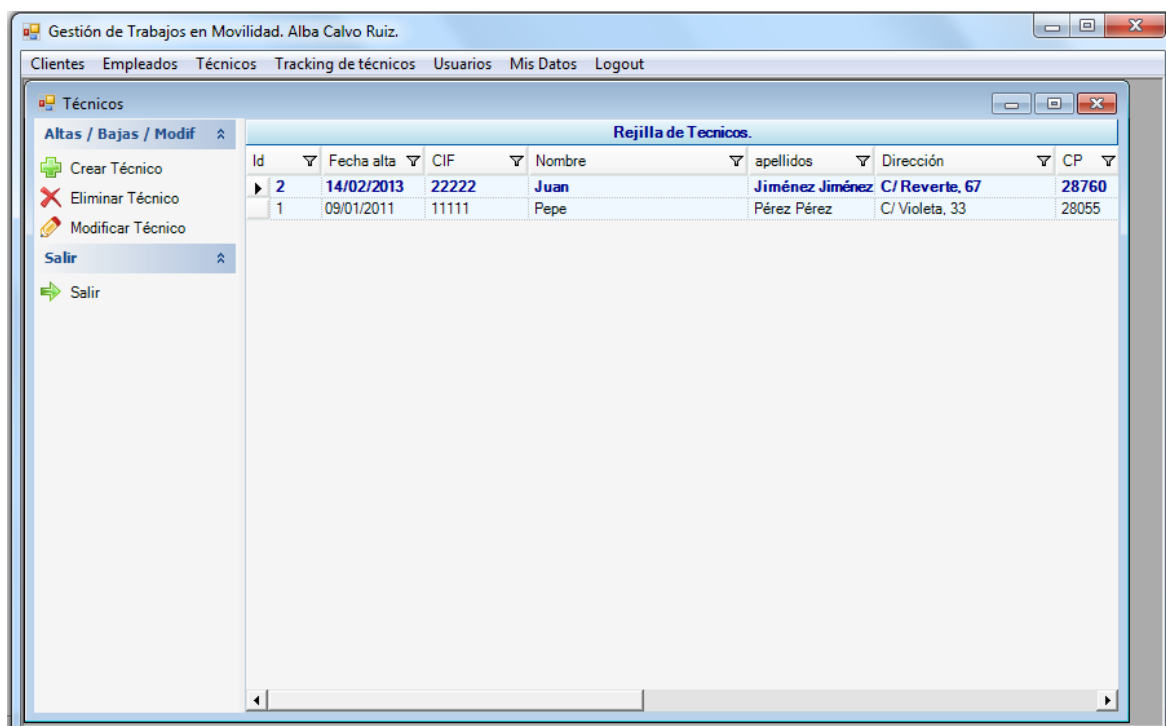


Figura 84. Anexo II: Manual de usuario. Aplicación de gestión: Ventana de Técnicos.

La selección de la opción “Técnicos” del menú principal abre el formulario correspondiente. El formulario presenta un listado de técnicos que podemos ordenar y buscar por varios criterios y además permite la realización de altas, bajas y modificaciones mediante las opciones presentes en la barra lateral izquierda.

### 3.2.4 Tracking de técnicos

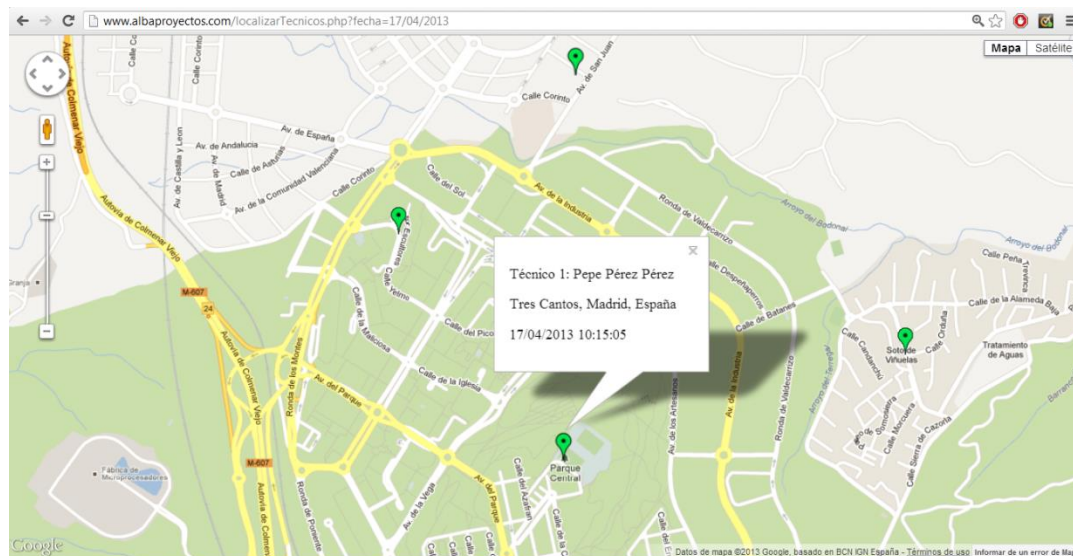


Figura 85. Anexo II: Manual de usuario. Aplicación de gestión: Ventana de tracking.

La selección de la opción “Tracking de técnicos” muestra una ventana que cuenta con dos opciones:

- Ubicación de técnicos:

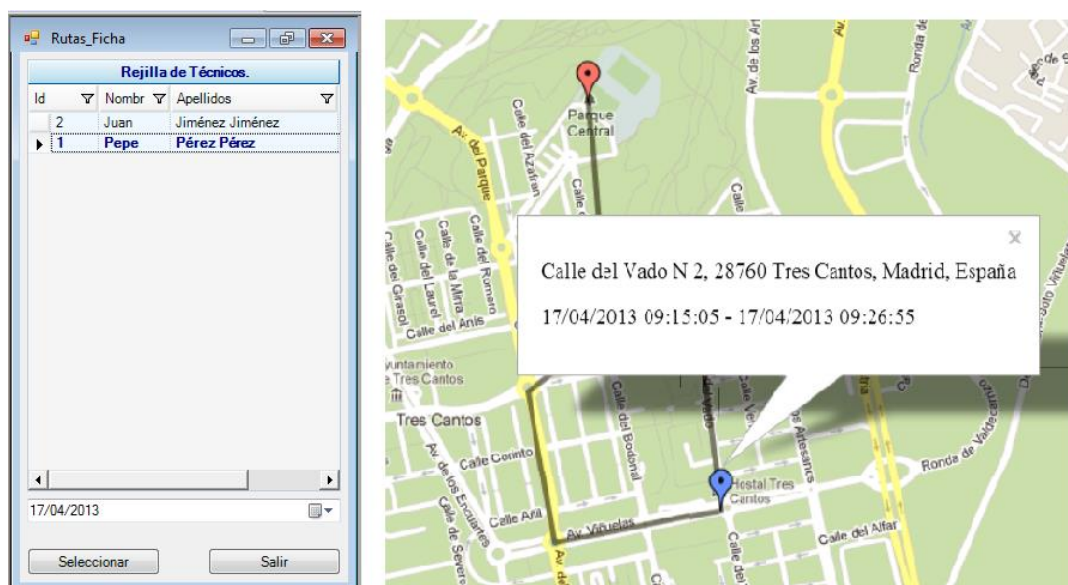
La selección de esta opción implica la apertura de una instancia del navegador predeterminado que le mostrará un mapa con la última posición conocida de todos los técnicos para la fecha actual mostrando la posición de cada técnico con un marcador verde.



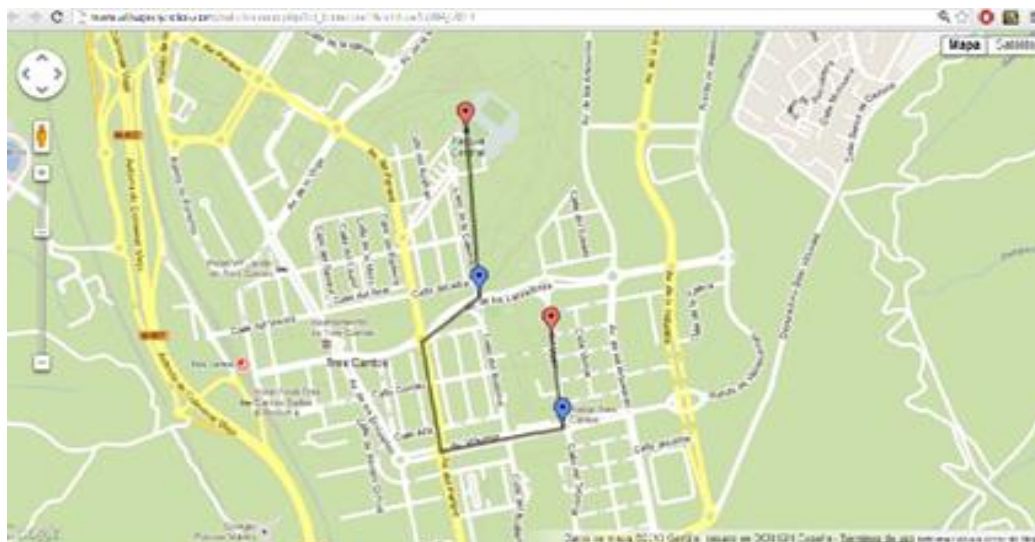
**Figura 86. Anexo II: Manual de usuario. Aplicación de gestión: Ubicación de técnicos.**

- Ruta de técnicos:

Esta opción le mostrará un formulario que solicita un técnico y una fecha. Tras introducir estos datos, se abrirá una instancia del navegador predeterminado y se mostrará un mapa del recorrido del técnico en esa fecha, representando el inicio y el fin del recorrido con un marcador rojo y las paradas del técnico (más de 5 minutos en un punto) en azul. Si se selecciona el marcador se mostrará la dirección y la hora de inicio y fin de la parada.



**Figura 87. Anexo II: Manual de usuario. Ruta de técnicos 1.**



**Figura 88. Anexo II: Manual de usuario. Aplicación de gestión: Ruta de técnicos 2.**

### 3.2.5 Usuarios

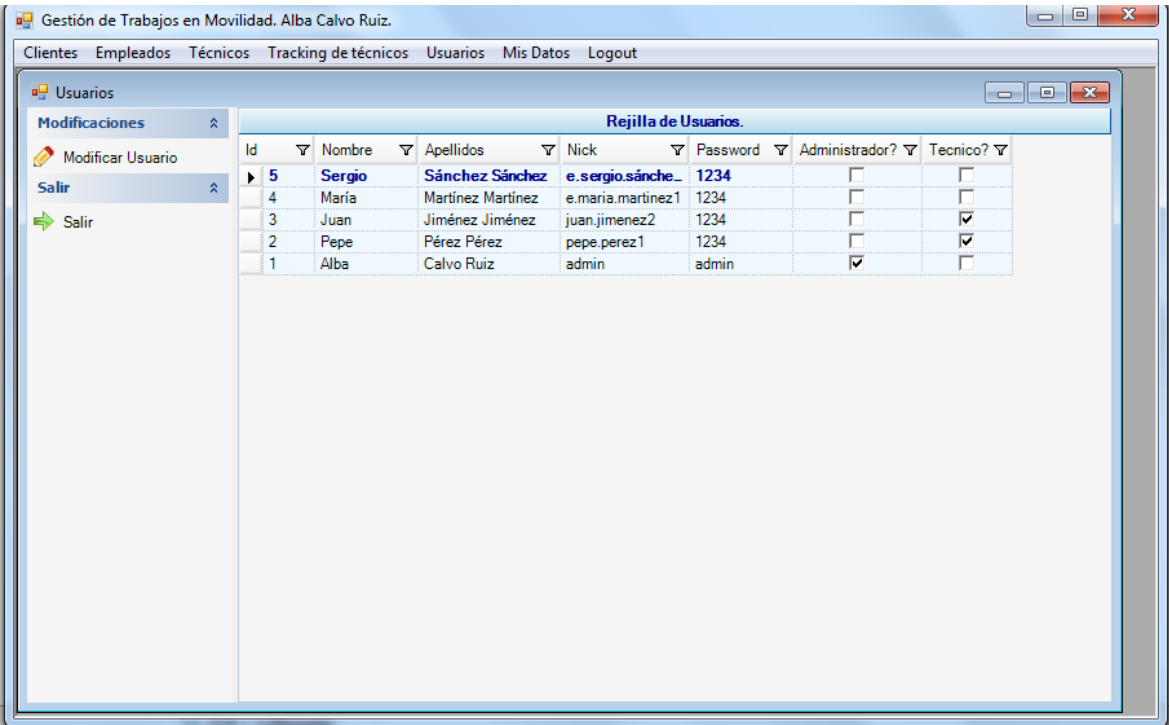


Figura 89. Anexo II: Manual de usuario. Aplicación de gestión: Ventana de Usuarios.

La selección de la opción “Usuarios” del menú principal abre el formulario correspondiente. El formulario presenta un listado de usuarios que podemos ordenar y buscar por varios criterios y además permite la realización de altas, bajas y modificaciones mediante las opciones presentes en la barra lateral izquierda.

### 3.2.6 Mis datos

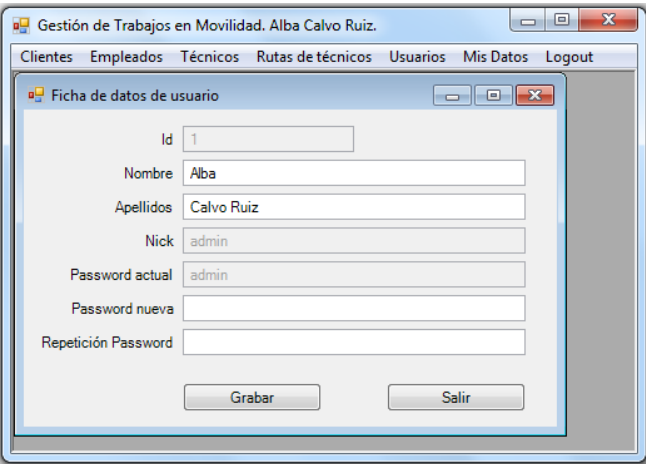


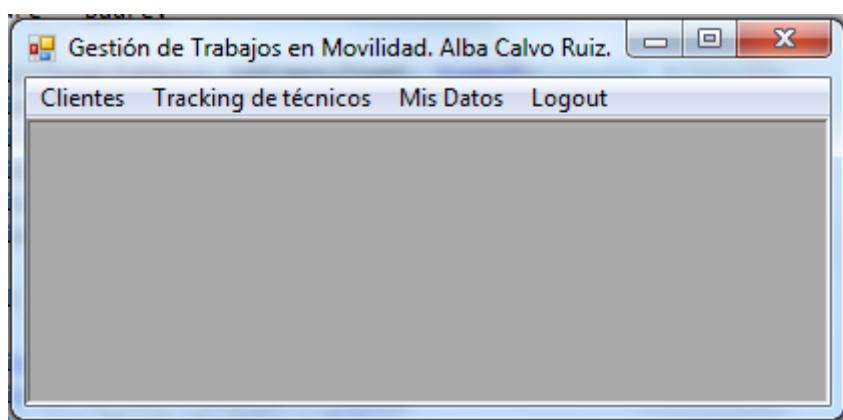
Figura 90. Anexo II: Manual de usuario. Aplicación de gestión: Ventana de datos personales de administrador.

La selección de la opción “Mis datos” del menú principal abre el formulario correspondiente. El formulario presenta sus datos de usuario y le permite modificar su nombre, apellidos y contraseña.

### **3.2.7 Logout**

La selección de la opción “Mis datos” del menú principal finaliza la sesión y retorna a la pantalla de login.

## **3.3 Opciones de empleado**



**Figura 91. Anexo II: Manual de usuario. Aplicación de gestión: Menú de empleado.**

Tras identificarse, se le mostrará un menú con varias pestañas que representan sus opciones como usuario empleado.

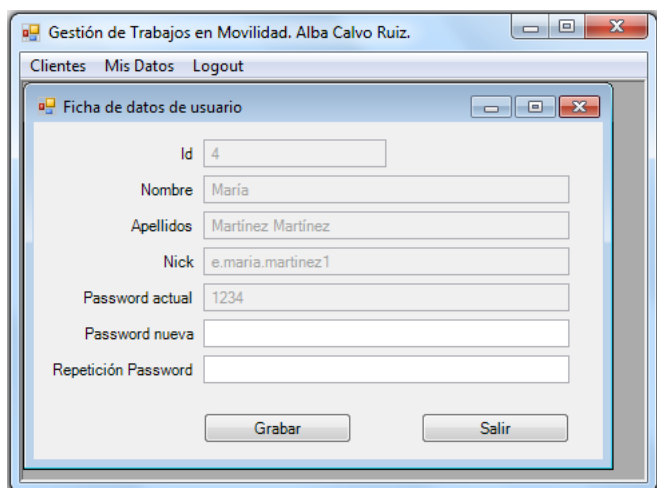
### **3.3.1 Clientes**

Por favor, consulte el apartado 3.2.1. Clientes.

### **3.3.2 Tracking de técnicos**

Por favor, consulte el apartado 3.2.4. Tracking de técnicos.

### **3.3.3 Mis datos**



**Figura 92. Anexo II: Manual de usuario. Aplicación de gestión: Ventana de datos personales de empleado.**

La selección de la opción “Mis datos” del menú principal abre el formulario correspondiente. El formulario presenta sus datos de usuario y le permite modificar su contraseña.

## 4 Manual de la aplicación Web

### 4.1 Pantalla de identificación



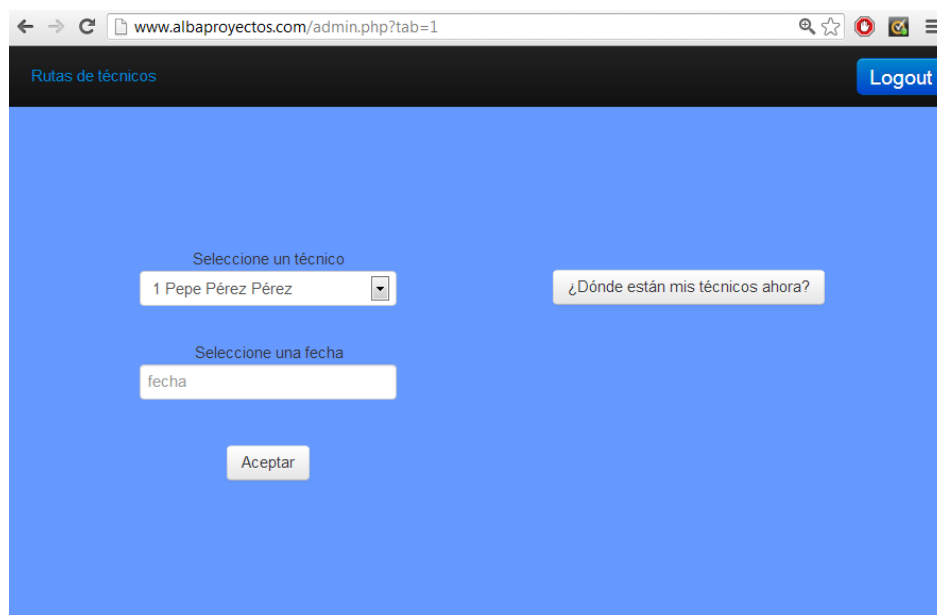
**Figura 93. Anexo II: Manual de usuario. Aplicación Web: Login.**

Lo primero que debe hacer para comenzar a utilizar la aplicación es autenticarse con sus datos de usuario.

La aplicación distingue dos tipos de usuario, administrador y técnico. Si usted tiene el primer rol, diríjase al siguiente apartado (4.2. Opciones de administrador). De lo contrario, diríjase al apartado 4.3 (Opciones de técnico).



## 4.2 Opciones de administrador

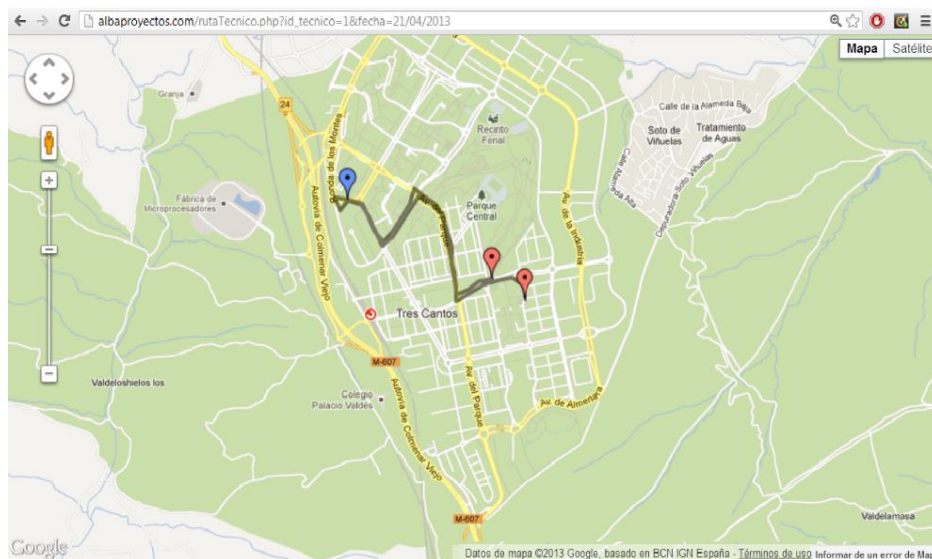


The screenshot shows a web browser window with the address bar displaying `www.albaproyectos.com/admin.php?tab=1`. The page has a dark blue header with the text "Rutas de técnicos" on the left and a "Logout" button on the right. The main content area is light blue and contains two sections. The first section, titled "Seleccione un técnico", has a dropdown menu showing "1 Pepe Pérez Pérez". To its right is a button labeled "¿Dónde están mis técnicos ahora?". The second section, titled "Seleccione una fecha", has a text input field containing the word "fecha". Below these sections is an "Aceptar" button.

**Figura 94. Anexo II: Manual de usuario. Aplicación Web: Menú de administrador.**

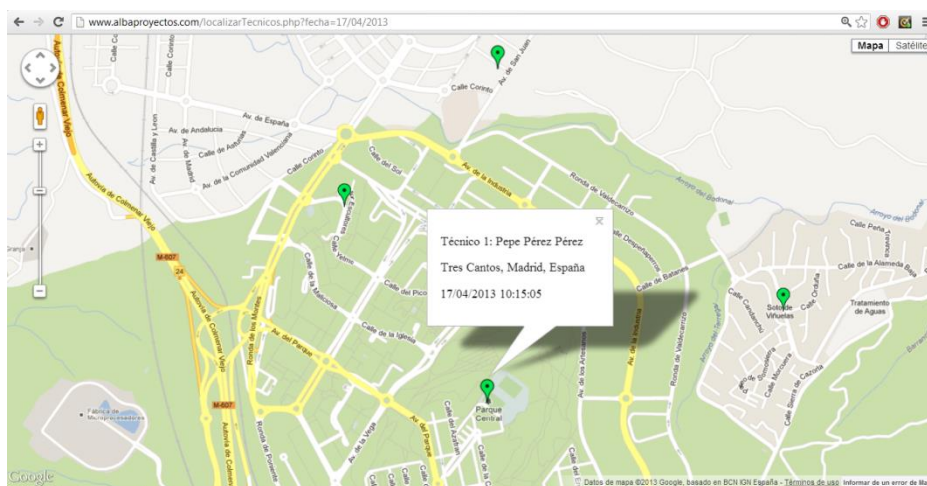
Tras identificarse, se le mostrará un menú dos opciones:

- Obtención de ruta seleccionando un técnico y una fecha (sección izquierda):



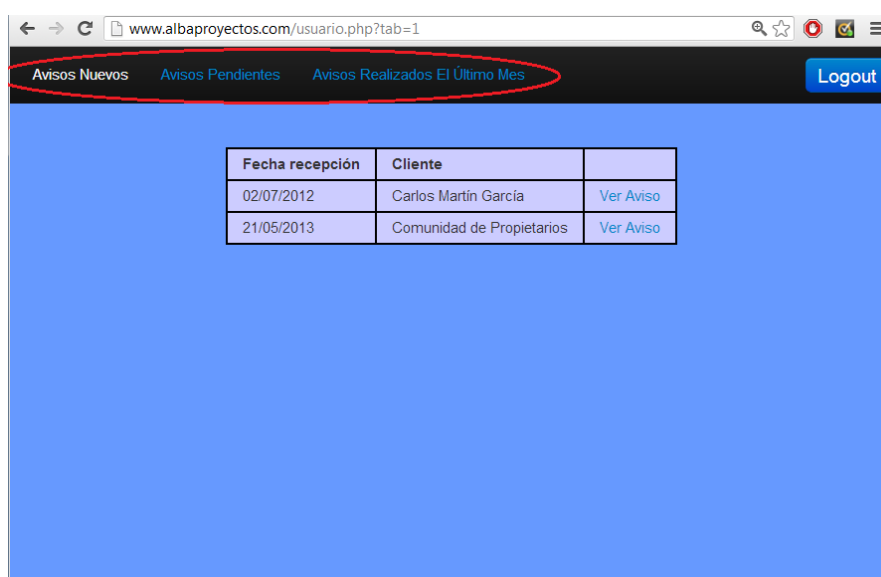
**Figura 95. Anexo II: Manual de usuario. Aplicación Web: Ruta de técnicos.**

- Obtención de ubicación de todos los técnicos de la empresa en este momento (sección derecha):



**Figura 96. Anexo II: Manual de usuario. Aplicación Web: Ubicación de técnicos.**

### 4.3 Opciones de técnico

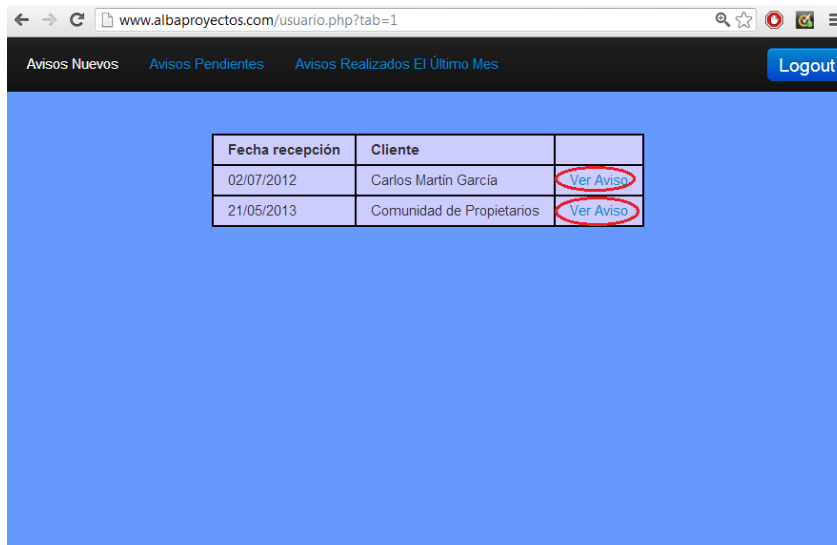


**Figura 97. Anexo II: Manual de usuario. Aplicación Web: Menú de técnico.**

Tras identificarse, se le mostrará un menú con tres pestañas: avisos nuevos, avisos pendientes y avisos realizados el último mes, que representan sus opciones como técnico.

Inicialmente la pestaña activa es avisos nuevos.

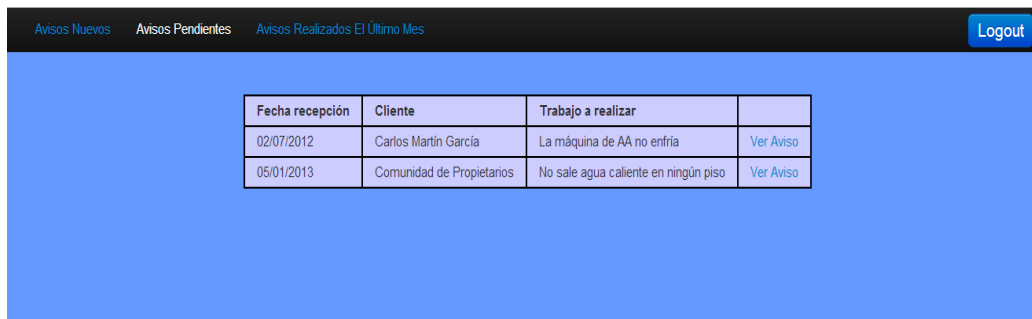
### 4.3.1 Avisos nuevos



**Figura 98. Anexo II: Manual de usuario. Aplicación Web: Avisos nuevos.**

Esta pestaña muestra los avisos que usted aún no ha examinado. Debe pulsar “Ver Aviso” para ver sus detalles y que sea marcado como visto y pueda ser realizado.

### 4.3.2 Avisos pendientes



**Figura 99. Anexo II: Manual de usuario. Aplicación Web: Avisos pendientes.**

Esta pestaña muestra una tabla como la del apartado anterior con el listado de todos sus avisos asignados que aún no han sido realizados. Dispone de nuevo de la opción “Ver aviso” para visualizar los detalles, pero ahora dispone de la opción “Anular/Finalizar” para cumplimentar y terminar el aviso con estado finalizado o anulado:.

**DATOS DEL AVISO**

Id aviso	7
Cliente	Comunidad de Propietarios
Dirección	C/ Mirlo, 12 Getafe, Madrid (28055)
Teléfono	91 3111111 91 3222222 633 111111
Fecha recepción	05/01/2013
Situación	PENDIENTE
Trabajo a realizar	No sale agua caliente en ningún piso

[Volver](#)
[Anular / Finalizar](#)

**Figura 100. Anexo II: Manual de usuario. Aplicación Web: Datos de aviso con opción de cumplimentación.**

**DATOS DEL AVISO**

Id aviso	7
Cliente	Comunidad de Propietarios
Dirección	C/ Mirlo, 12 Getafe, Madrid (28055)
Teléfono	91 3111111 91 3222222 633 111111
Fecha recepción	05/01/2013
Situación	PENDIENTE
Trabajo a realizar	No sale agua caliente en ningún piso

Trabajo realizado:

Reparación del calentador de agua.

Situación REALIZADO

[Aceptar](#)
[Cancelar](#)

**Figura 101. Anexo II: Manual de usuario. Aplicación Web: Cumplimentación de aviso.**

### 4.3.3 Avisos realizados el último mes

<a href="#">Avisos Nuevos</a>	<a href="#">Avisos Pendientes</a>	<a href="#">Avisos Realizados El Último Mes</a>	<a href="#">Logout</a>
-------------------------------	-----------------------------------	---	------------------------

Fecha finalización	Cliente	Trabajo realizado	
08/05/2013	Comunidad de Propietarios	Reparación del calentador de agua.	<a href="#">Ver Aviso</a>

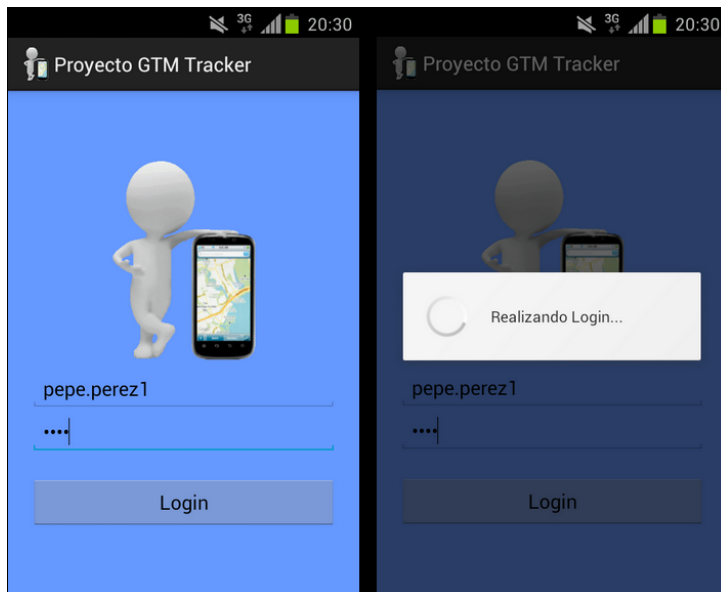
**Figura 102. Anexo II: Manual de usuario. Aplicación Web: Avisos realizados el último mes.**

Esta pestaña muestra una tabla con todos los avisos realizados por usted este mes. Sus detalles pueden ser consultados mediante la opción “Ver Aviso”.

## 5 Manual de la aplicación de geolocalización

---

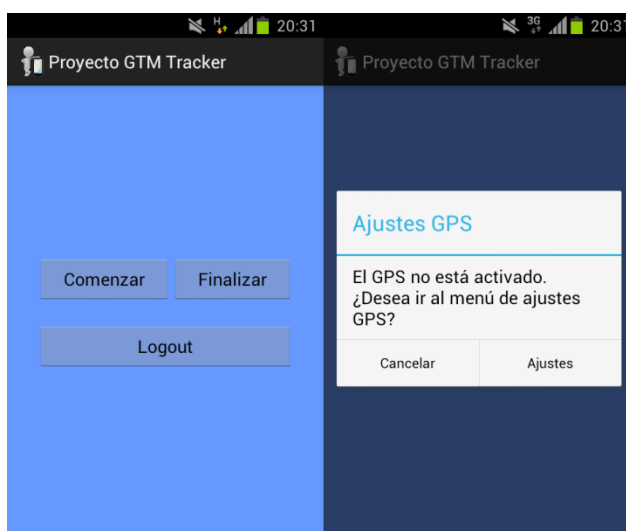
### 5.1 Pantalla de identificación



**Figura 103. Anexo II: Manual de usuario. Aplicación Android: Login.**

Lo primero que debe hacer para comenzar a utilizar la aplicación es autenticarse con sus datos de usuario.

### 5.2 Pantalla de geolocalización




**Figura 104. Anexo II: Manual de usuario. Aplicación Android: Menú / Habilidad de GPS.**

Debe asegurarse de que el GPS se encuentra habilitado. Si no lo está, la aplicación le mostrará un mensaje y podrá activarlo o bien pulsando “Ajustes” y accediendo al menú de

Android o mediante el descenso de la barra deslizante superior y la elección del icono de GPS. Ambas opciones se muestran en la siguiente imagen:



**Figura 105. Anexo II: Manual de usuario. Aplicación Android: Habilitación de GPS.**

A continuación debe iniciar el servicio de geolocalización pulsando el botón “Comenzar” y comprobar que en la barra superior aparece un icono parpadeante como el siguiente: 

Debe asegurarse de no hacer click en el botón “Finalizar” o “Logout” hasta el fin de su turno de trabajo para que su posición pueda enviarse correctamente.